UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

MASTER THESIS

Abandoned object detection in long-term video-surveillance

Master's Degree in ICT Research and Innovation

Image Processing and Computer Vision Program

Elena Luna García
Director: Juan Carlos San Miguel Avedillo
Supervisor: Jose María Martínez Sánchez

July 2017

Abandoned object detection in long-term video-surveillance

Elena Luna García
Director: Juan Carlos San Miguel Avedillo
Supervisor: Jose María Martínez Sánchez

Video Processing and Understanding Lab
Escuela Politécnica Superior
Universidad Autónoma de Madrid
July 2017

This work has been partially suported by *Ministerio de Economía, Industria y Competitividad* of the Spanish Government and *Fondo Europeo para el Desarrollo Regional* of the European Union under the project TEC2014-53176-R (HAVideo)

Abstract

Due to recent events, global security concern is significantly increasing in our society. This area of concern is directly connected with the growing demand of video surveillance systems, mainly in public and crowded scenarios, such as railway stations, due to the potential risk they present.

In order to avoid the arduous manual task of supervising a video surveillance system, automatic analysis and detection of this kind of events is the challenging task to be achieved. Although diverse systems trying to reach this goal have been proposed in the literature there is a lack of evaluation within this field.

An end-to-end configurable system for abandoned and stolen object detection has been designed and developed by integrating available techniques. This systems integrates several algorithms in each module of the system, thus it allows the evaluation of several state-of-the art techniques combinations. An evaluation protocol considering short and long-term sequences has been designed by classifying available datasets and analysis evaluation metrics.

A graphical user interface has been developed allowing the algorithms and parameters selection and adjustment for each stage of the system, as well as displaying the results.

In addition, a new different system integrating recent and innovative state of the art proposals has been proposed.

Keywords

Video analysis, abandoned object detection, stolen object detection, static objects detection, video-surveillance.

Resumen

Debido a los acontecimientos recientes, la preocupación por la seguridad global está incrementando en nuestra sociedad. Este motivo de preocupación está directmente realacionado con la creciente demanda de sistemas de vídeo vigilancia, principalmente en espacios publicos y transitados, tales como estaciones de tren, debido al potencial riesgo que presentan.

Con el fin de evitar la ardua tarea manual de supervisar un sistema de video vigilancia, surge la difícil tarea de analizar y detectar automáticamente eventos como el robo y abandono de objetos. Aunque se han propuesto diversos sistemas tratando de alcanzar esta meta, aún se carece de un marco de evaluación en este área.

En este trabajo se ha diseñado y desarrollado un sistema completo configurable para la detección de eventos abandonados y robados integrando las técnicas disponibles en el estado del arte. El sistema integra diferentes algoritmos en cada uno de sus módulos, haciendo posible la evaluación de diferentes combinaciones. Se ha diseñado también un protocolo de evaluación para secuencias de corto y largo plazo, mediante la clasificación de las mismas y el análisis de métricas de evaluación.

Se ha desarrollado una interfaz gráfica de usuario que permite la selección y ajuste de los algoritmos y parámetros para cada módulo, así como la visualización de los resultados de cada etapa.

Además, se ha propuesto un nuevo sistema que integra diferentes técnicas recientes del estado del arte.

Palabras clave

Análisis de vídeo, detección de objetos abandonados, detección de objetos robados, detección de objetos estáticos, vídeo vigilancia.

Agradecimientos

En primer lugar, quisiera agradecer a Juan Carlos San Miguel, mi tutor y guía a lo largo de este trabajo, prestarme su atención y ayuda siempre que han sido necesarias.

Agradecer también a Diego, Rafa y Álvaro haber estado siempre dispuestos a ayudarme a solucionar los problemas que han ido surgiendo a lo largo de estos meses.

Quiero dar las gracias a quienes han sido como mi familia durante los dos últimos años. A Marta, Álex, Iñaki y Erik por ser valientes y lanzarse a la aventura. A Marta, por comprenderme. A Álex, por ser uña y carne y por su ayuda inestimable. A mi nueva familia húngara y francesa, por haber hecho única esta experiencia, sin duda.

Gracias a Adriana, mi Nanis, por apoyarme tanto en la lejanía como en la cercanía y sobreestimarme siempre.

Gracias a mi familia, por recorrerse Europa para verme y estar al pie del cañon día a día.

De corazón, gracias.

Elena Luna García.

2017.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Nowadays, due to recent events, the global security concern is significantly increasing in our society. This concern is directly connected with the growing demand of video surveillance systems [1], mainly in public and crowded scenarios, such as airports, buildings or railway stations [2]. Attention is focused in these places because they present a potential risk for dangerous situations such as object abandonment and/or important objects stealing.

Traditionally, video surveillance systems' monitoring task has been performed by human operators analysing information coming from several cameras which is displayed simultaneously. For this reason it is anticipated that the operator's attention span will decrease as the information to process increase. In order to avoid this arduous manual supervising task, automatic analysis and detection of events of interest is the challenging task to be achieved, in Figure 1.1 two examples of automatic detection warnings are shown. In this area, stolen and abandoned objects, as well as parked vehicles, detection has become an active research topic.

Various systems trying to reach this goal have been described in the literature [3, 4, 5]. Typically, abandoned and stolen objects detection is achieved by the development of a video analysing system that is formed by the following stages: background segmentation, stationary foreground detection, person/object discrimination and finally, stolen/abandoned classification. This modules are described in Chapter 2.

The developed applications are required to perform correctly under complex scenarios such us crowded sequences or changing situations. At present, this is a partially solved issue as each stage of the system has to deal with several challenges affecting

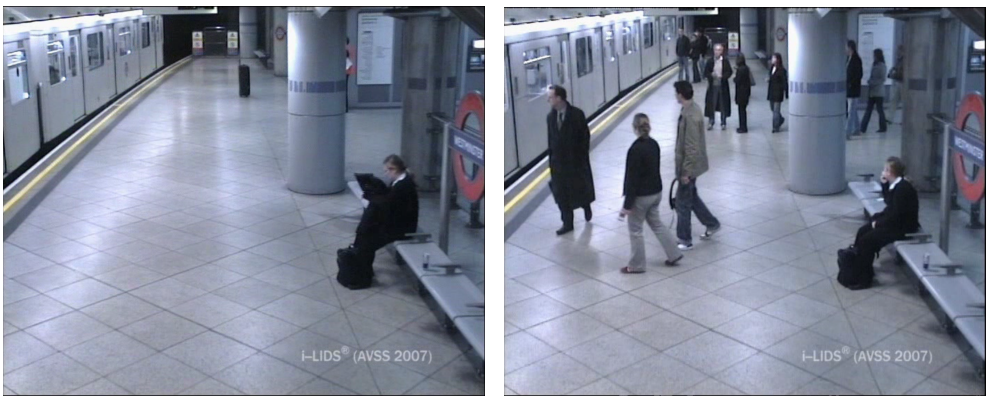
(a) Abandon object alert (b) Removed object alert

Figure 1.1: Sample images of detected abandoned and removed objects.

Figure 1.2: Sample images of illumination changes over time in the same scene, from [6].

its performance. Multimodal backgrounds and illumination changes (see Figure 1.2, for an example) can result in a wrong background segmentation, which will complicate the static foreground extraction. Regarding the static foreground detection stage, its main challenge are the environments with high density of objects because of their motion speed and high number of occlusions (an example of an occluded abandoned object is shown in Figure 1.3). Abandoned and stolen objects can be shaped in several and arbitrary ways and colors and this appearance variability can make the classification task between objects and people harder. Long-term is also one of the main challenges in abandoned and stolen object detection in video surveillance, we can consider long-term videos as sequences that last long enough to involve challenges such as illumination changes along the sequence, object occlusions or changes in the scene context. As it will be shown in next chapter, each stage is totally reliant on the previous one, therefore, if an error arises it will be carried along the entire system. It is also noteworthy that, ideally, the real time analysis is a desired goal in applications for security/monitoring staff support.

Some works have been presented in the literature trying to deal with the mentioned



(a) Unattended luggage (b) Occluded unattended luggage

Figure 1.3: Sample images of an occluded unattended bag.

problems [7, 8], but they all evaluate in each system stage a single technology they consider the best candidate, therefore, there are several technologies that have not been proven yet. This lack of an evaluation framework have motivated this work. The second motivation was the need to evaluate long-term video surveillance sequences due to the challenges they present and the absence of an evaluation in the literature.

1.2 Objectives

This Master Thesis will focus on the detection of abandoned and stolen objects in long-term video surveillance. The main objective is to develop an end-to-end system for abandoned and stolen object detection which will be thoroughly evaluated and to contribute to the existing approaches for selected analysis stages of such system.

The main objective mentioned above can be split in several subgoals:

1. Study and comprehension of the state of the art available techniques

   Attention will be focused on techniques for abandoned and stolen object detection in video sequences. In particular, background subtraction, stationary foreground detection, people detection and classification stages will be theoretically studied, as well as the available datasets.

2. System design and development

   To design and develop a complete configurable system based on a previous analysis of the available algorithms in the VPU-Lab (http://www-vpu.eps.uam.es). For this purpose some available algorithms will be adapted and integrated into the system, making use of the OpenCV library.

3. System improvements

   Some innovative proposals will be presented and implemented to improve previous results.

4. Evaluation protocol

   To design and develop an evaluation protocol by classifying available datasets, and analysing evaluation metrics. The obtained results will be compared with the state of the art results so a comparison can be made.

5. Development of a graphical user interface

   To design and develop a graphical user interface allowing the algorithms and parameters selection and adjustment for each stage of the system, as well as displaying the results.

1.3 Thesis Structure

The master thesis report is divided into the following chapters:

- Chapter 1. Introduction.
- Chapter 2. State of the Art.
- Chapter 3. Configurable system.
- Chapter 4. Proposed system.
- Chapter 5. Evaluation methodology.
- Chapter 6. Results.
- Chapter 7. Conclusions and Future Work.
- References.

Chapter 2

State of the Art

In this chapter the state of the art related to stolen and abandoned objects detection in video surveillance will be studied. This chapter is divided into the following sections: video surveillance systems introduction in Section 2.1, foreground segmentation definition and its more important techniques in Section 2.2, stationary foreground detection definition and the most relevant techniques in Section 2.3, foreground classification in Section 2.2, and lastly, stolen/abandoned classification definition and its techniques in Section 2.5.

2.1 Introduction

Video surveillance systems were originally composed of a closed circuit television (CCTV), as it is described in [9]. In a closed circuit television system all the elements such as camera, display monitors, recording devices, etc. are directly connected with each other in a closed circuit. The video signal, although it was a digital signal, was initially transmitted as analogue through coaxial cable to the monitoring room, equipped with a monitor matrix, where the signal was displayed. This kind of systems are called as the first generation video surveillance systems.

The second generation systems took advantage of the digital nature of the signal and introduced signal processing. In these systems the signal was not convert into analogue, but it is received directly on a computer that processes it by applying some image analysis algorithm. Most of these algorithms were real time event detection algorithms, helping the human supervisor to detect incidents. The disadvantage these systems present is that powerful computers are needed in order to process a large number of cameras.

The aim of the third generation was to design a distributed architecture, instead

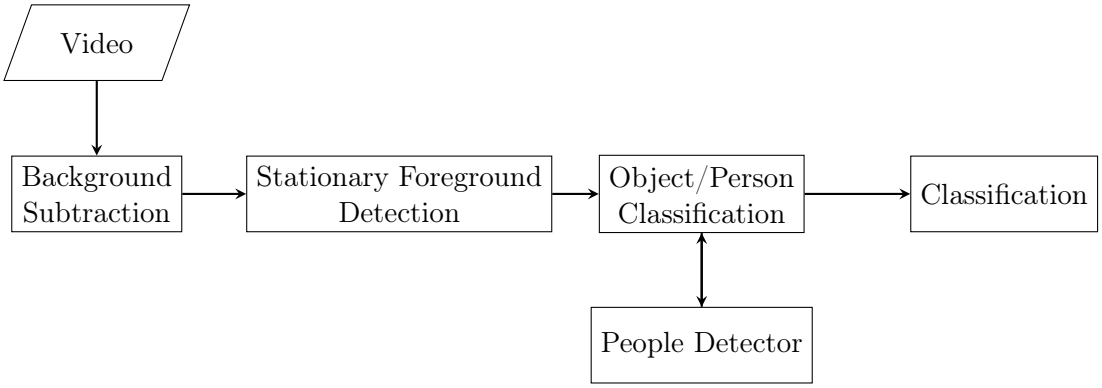


Figure 2.1: Stolen and abandoned object detection system block diagram.

of the centralised architecture of the previous generation, capable of dealing with multiple cameras, a geographical spread of resources and many monitoring points. The purpose of these systems is to exploit automatic video understanding technologies in order to allow a single person to monitor a complex sequence. From an image processing point of view, they are based on the distribution of processing capacities over the network and the use of embedded signal processing devices to give the advantages of scalability and robustness potential of distributed systems [10].

Table 2.1 shows a surveillance systems summary in terms of the main problems and current researches of the three generations.

	First Generation	**Second Generation**	**Third Generation**
Technique	Analog CCTV systems	Combine computer vision technology with CCTV systems for automatic video surveillance	Automatic broad area surveillance system
Problem	Utilize analog techniques for image distribution	Robust object detection and tracking algorithms to recognize e-vents	Distribution of information
Research	CCTV video compression	Real-time robust computer vision algorithms	Multi-camera surveillance techniques

Table 2.1: Evolution of the surveillance systems, from [9].

Typically, the systems proposed in the literature follows a scheme as the one showed in Figure 2.1, including several stages. In the following sections all these stages will be analysed in more detail.

(a) Current frame (b) Background model (c) Foreground mask

Figure 2.2: Current frame, background model and foreground frame examples.

2.2 Foreground segmentation

2.2.1 Description

Foreground segmentation is the first step in many computer vision applications. It consists in locating objects of interest within a scene through the distinction between background and foreground pixels. There are several different video segmentation techniques depending on which kind of sequences will be analysed and the results that are expected to be achieved. Most of these techniques make use of background subtraction algorithms in order to obtain the regions of interest. Background subtraction algorithms obtain a binary foreground-background mask through a comparison between the image under analysis (I_t) and a background model (B_t):

$$FG_t = f(I_t, B_t)$$

Out of all methods presented in the literature, pixel-level algorithms are the ones most commonly used. Examples are computing the background by averaging out the analysed frames, by computing the median of each pixel within a n previous frames buffer or by modeling each pixel with a probability density function. Figure 2.2 shows the background model and the foreground mask generated by a background subtraction algorithm.

A key parameter of background subtraction algorithms is the learning rate. This rate defines how the algorithm will adapt the background changes, for instance, a low learning rate will produce a wide model with difficulty in detecting a sudden change in the background. On the contrary, if the model adapts too quickly, slowly moving foreground pixels will be absorbed into the background model, impeding the algorithm to detect them as foreground. For this reason this is a critical stage, since the following stages results are completely dependent on this one. Foreground segmentation stage

has to deal with multiple complications, therefore it is one of the hardest and trickiest tasks within the scope of video surveillance signal processing. This is due to the fact that the foreground detection algorithms developed so far are very sensitive to sudden illumination changes, weather conditions, background update, shadows and reflects, camouflage (similarities between object and background), noise and/or multimodal backgrounds [11, 12]. Illumination changes or reflects will provide false positives, i.e. they will be wrongly detected as a stolen or abandoned object.

The main algorithms used for this purpose are described down below.

2.2.2 Algorithms

Main foreground segmentation techniques in the literature are shown in Table 2.2 beside their robustness, level of analysis and features taken into account.

- **GMM (Gaussian Mixture Model)**. GMM is a parametric model based on characterising each pixel of an image with a mixture of Gaussians [13, 14, 15, 16]. Each pixel is represented by the weighted sum of K Gaussians, where K is a parameter to define, and each Gaussian is described in terms of each pixel color component mean and standard deviation.

- **HFSM + SVM (Hierarchical Finite State Machine + Support Vector Machine)**. It is a non-parametric method based on characterising each pixel of an image through a hierarchic three-layer state machine (pixel, region and event layers) [17, 18]. The pixel layer is composed of three pixel states: b, f and s (background, foreground and stationary, respectively), and it determines each pixel state via intensity and time features. Each pixel is considered as background during the background modeling thus$b$ is the initial state.

- **ESD (Edge-Segment Distribution)**. This non-parametric method characterises each pixel of an image through the information extracted from its edges. As it is explained in [19], this method is based on a foreground detection method [20], which creates edge-segment distributions from a training sequence as a background and incoming frames. Color and gradient information is added to the background to disambiguate foreground edges that may be confused with background. Additionally, an unknown object map is created from the temporal model to group the edges and detect the unattended objects.

- **SOM (Self-Organized Model)**. It is a non-parametric, multimodal, recursive, and pixel-based method able to learn the background through a map of motion and stationary patterns, making use of neuronal maps [21, 22, 23].

Technique	BGS	Robustness			Analysis	Features
		Illumination changes	Occlusions	Shadows/ Reflects		
GMM	Yes	Yes	Yes	Yes	Pixel	Color, brightness and texture
HFSM + VSM	No	Yes	Yes	No	Pixel and region	Area, intensity, shape and speed
ESD	Yes	Yes	Yes	No	Pixel borders	Borders
3D-SOBS(SOM)	Yes	Yes	No	Yes	Pixel	Brightness
KDE	Yes	Yes	Yes	No	Pixel and region	Depth
Color + Struct. Diff	No	Yes	No	Yes	Pixel and region	Color and borders

Table 2.2: Main foreground segmentation techniques comparison

- **KDE (Kernel Density Estimation)**. KDE is a non-parametric method able to store the most recent values of each pixel and estimate the density function of its distribution allowing to know its probability of being part of the foreground on a quick manner [24, 25].

- **Color + Structural Difference**. This method proposes an hybrid differencing. A color difference between the frame and current background is computed for each pixel, and in order to make it robust to local illumination changes a structural difference of the local patch around the pixel is also computed [26]. Finally, the difference image resulted from hybrid differencing is thresholded and connected component analysis is performed to generate candidate abandoned and stolen blobs.

Currently, GMM is most commonly used model due to its capability to model background variations such as gradual illumination changes, shadows and repetitive movements [27, 28].

2.3 Stationary foreground detection

2.3.1 Description

Once the foreground is determined, stationary foreground detection is the next step in abandoned and stolen objects detection systems, Figure 2.1. It merely consists in identifying which objects in the scene remain static. This stage should improve the foreground data quality in order to detect correctly the static regions. It plays a very important role because the abandoned/stolen classification will be carried out over the stationary objects detected, which means that missed and false detections in this stage will lead to wrong results at the end.

(a) Current frame (b) Foreground binary mask (c) Static foreground binary mask

Figure 2.3: Static foreground mask, in 2.3c, extracted from the foreground mask, in 2.3b, obtained by applying a background subtraction algorithm over the current frame, in 2.3a, belonging to CANDELA dataset, available at www.multitel.be/image/research-development/research-projects/candela/abandon-scenario.php.

Within detection stationary foreground detection scope, methods based on background subtraction have become very popular due to the common use of cameras that are fixed and the assumption that the illumination changes in the scene occur in a gradual way [29, 3] For this reason, in this work we will only focus in stationary foreground detection based on background subtraction. In Figure 2.3 an example of this approach is shown.

2.3.2 Algorithms

Within the scope of stationary foreground detection algorithms based on background subtraction, two main groups can be considered: those using object tracking [30, 31] and those only making use of analysing techniques [32]. Approaches using object tracking are suitable for low object density scenarios and they are widely used. Furthermore, approaches based on analysing the foreground mask are suitable for high object density scenarios and they require less computational cost. For these reasons this work will only consider approaches based on analysing the foreground detection stage results without tracking.

In [33] a classification of the background subtraction based methods for stationary regions detection without using tracking algorithms is proposed. As can be seen in Figure 2.4, they are divided into two main categories depending on how many background subtraction models they use: using just a model or using several. These two categories are both, in turn, divided into two categories depending on the processing frame rate: frame-by-frame analysis and sub-sampled analysis at different rates.

The main difficulty while detecting static regions of interest lies in the fact that objects appearance present large variations due to viewpoint changes, scene disorder,

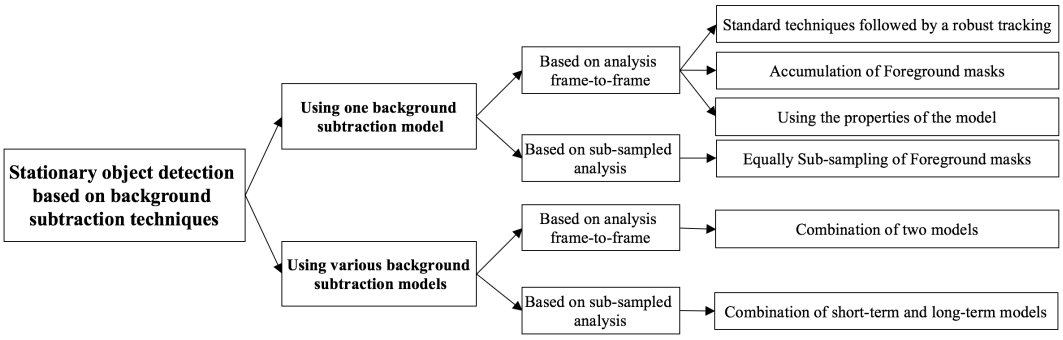


Figure 2.4: Classification of the background subtraction based methods for stationary object detection, from [33].

illumination changes and unlimited colors and shapes, which means that, for instance, the same object appearance may vary under different conditions. Below, some state of the art stationary foreground detection techniques are described.

- **Foreground mask sampling** [34]. This approach identifies the regions of interest in the foreground mask, i.e. static foreground, by logical foreground-background reasoning without taking into account appearance information. For this reason, this approach can deal with objects of arbitrary shape and color without the need for prior learning. It also works properly in crowded and highly-cluttered scenarios. Within the classification in Figure 2.4 scope, this is a method based on equally sub-sampled analysis using one background subtraction model.

- **Accumulation mask (ACC)**. This method is described in [35]. It computes a stationary region confidence map, where each pixel of the image represents the confidence of that pixel being part of a stationary object. An increment counter is used when a pixel does not fit with the background, and a decrement counter is used when a pixel fits with the background. The confidence map is updated every frame using these counters and finally it is thresholded to obtain a binary mask with the static foreground. Regarding Figure 2.4, this is a method based on frame-by-frame analysis using one background subtraction model.

- **Motion Filtering + ACC [16]**. This approach combines foreground accumulation mask method, described above, with a motion filtering in order to add robustness against high object density scenes.

- **Dual background model + ACC (DBM + ACC)** [36]. Dual background model approaches make use of two, long-term and short-term, background mod-

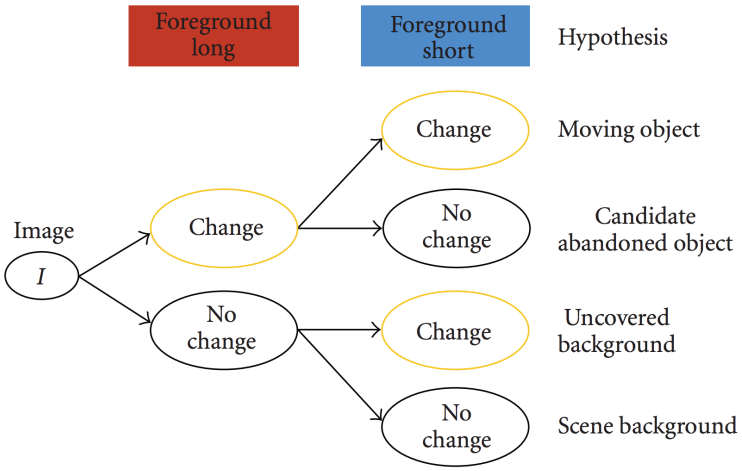


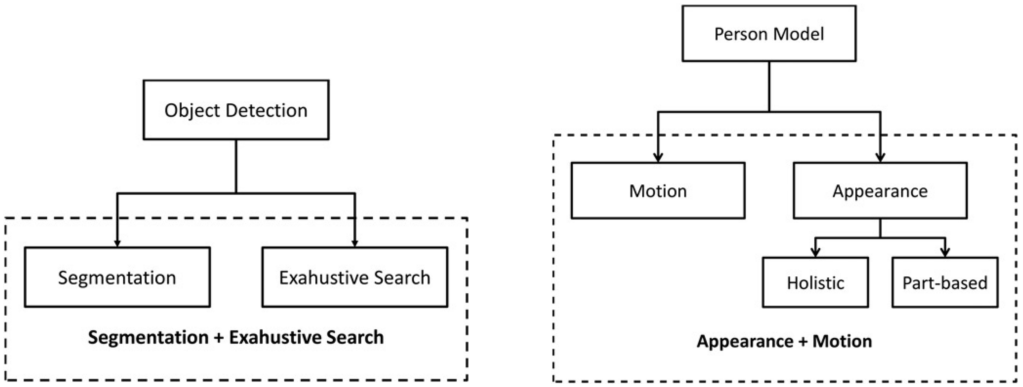
Figure 2.5: Hypotheses on long and short-term foregrounds, from [36].

els. Long-term background model is updated every n frames, presenting a low learning rate with respect to adapting changes in the scene. On the other hand, short-term background presents a higher learning rate and it adapts quickly to changes because it is updated frame by frame. By this way, two foreground masks are obtained at each frame and depending on theirs values the hypotheses shown in Figure 2.5 are considered. Finally, the detection results are incorporated into a confidence image by updating the pixel values in a similar way to accumulation mask methods.

2.4 Object/person classification

Next stage consists in classifying the stationary regions, detected in the previous stage, in objects and people. Within this process a simple assumption can be made: whatever is not a person can be considered as an object. Following this strategy, this problem can be solved by applying a people detector. Although it may seem a simple task, two complexities make it a challenging task:

- Great variability in people appearance. This variability is due to multiple different clothes people can wear, personal belongings (bags, umbrellas, etc.) and the large number of poses a person can adapt (walking, standing, sitting, bending, etc.).
- Computational time. Real time computation is a really challenging task due to the fact that complex people detectors require high computational cost, which entails high computational time.



(a) People detection classification regarding the object detection task

(b) People detection classification regarding the person model

Figure 2.6: People detection approaches classification, from [38].

Mostly people detection approaches consist, first, of designing and training (if required) a person model based on features and secondly, adjusting this person model to the candidates of being person in the scene, as described in [37]. There are two main critical tasks in every people detector approach: the object detection, consisting of extracting from the scene the object candidates to be a person, and the person model, which defines the characteristics and behaviour an object must have in order to be considered as a person. All candidates adjusting to the model will be considered as person, while the others will not do it. Two classifications regarding these two tasks are shown in Figure 2.6.

A large number of people detector algorithms are proposed in the literature [39] however, the most popular approaches, are those whose person model defines the people appearance according to their edge information using some shape descriptors, such as Haar-like features [40], Histogram of Oriented Gradients (HOG) [41, 42], Edgelets [43] or a combination of features, such as Aggregated Channel Features (ACF), introduced in [44]. These approaches are based on appearance because it is more discriminant than motion, however, as it is explained before, human appearance presents a great variability and it can also change because of external factors such as light conditions, clothing, contrast etc.

(a) Stolen object scenario.

(b) Abandoned object scenario.

Figure 2.7: Examples extracted from CANDELA dataset, available at www.multitel.be/image/research-development/research-projects/candela/abandon-scenario.php.

2.5 Stolen/abandoned classification

2.5.1 Description

The aim of this module of the system is to classify the stationary regions classified as objects in the previous stage in two categories:

- **Stolen object**. This category includes objects that originally were stationary in the scene and then are removed or changed (e.g theft or vandalism).

- **Abandoned object**. This category includes objects that originally were not in the scene and then are detected as stationary (e.g left luggage, illegal parked vehicle, etc).

Figure 2.7 shows two example situations of these categories. In Sub-Figure 2.7a one can observe a man removing a waste bin that was initially in the scene and in Sub-Figure 2.7b a man leaves a briefcase on the couch.

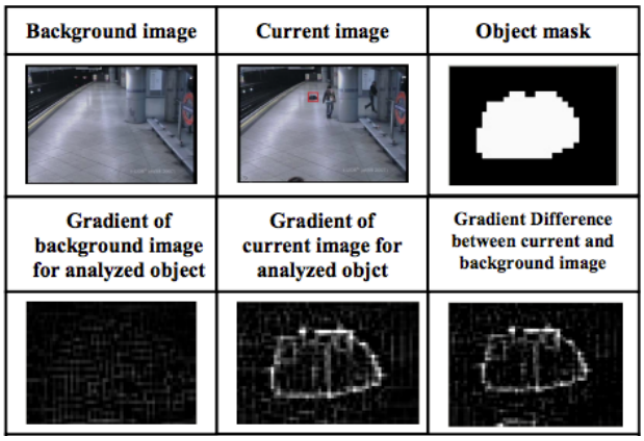


Figure 2.8: Example of High gradient detector for abandoned object with AVSS2007_Medium sequence, from [30].

2.5.2 Algorithms

Approaches proposed in the literature for stolen/abandoned classification can be classified, according to the features used to discriminate into edge-based, color-based and hybrid approaches [45].

- **Edge-based approaches** [46, 47]. These methods consider the energy of the region boundaries of the static object and makes the assumption that, in the current frame, it is higher for abandoned objects and lower for stolen objects. This reasoning can be observed in Figure 2.8, where an example of this method is shown. It is an example of an abandoned object and it is easy to see that the energy of the static object boundaries is higher in the current frame than in the background image.

- **Color-based approaches** [48, 7]. These methods are based on analysing the color information of the internal and external regions demarcated by the bounding box and the boundaries of the static object. An example of this method is shown in Figure 2.9.

- **Hybrid approaches** [30]. These are a combination of the previous approaches. They combine energy edge and color information together in order to determine the object nature.

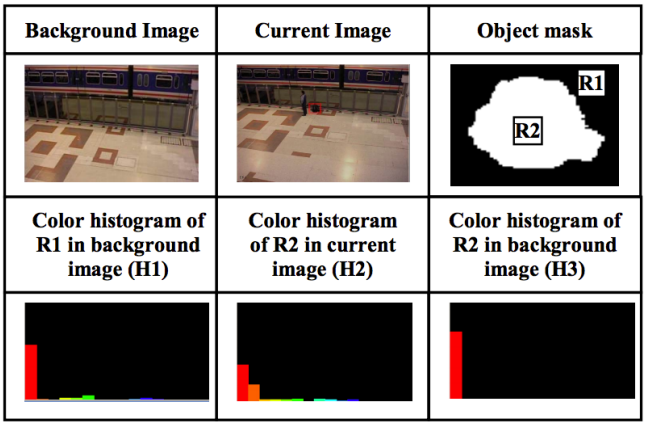


Figure 2.9: Example of color-based detection for abandoned object for PETS2006_S1 sequence, from [30].

Chapter 3

Configurable system overview

In this chapter the developed video analysis system for abandoned and stolen object detection is described. At first, in Section 3.1, the preliminary work is detailed. Section 3.2 describes the proposed system architecture and each module architecture and finally in Section 3.3 the designed and developed graphical user interface is described.

3.1 Adaptation to OpenCV 3.2

The system presented in this work has been developed from an initial prototype available at the Video Processing and Understanding Lab (VPULab) [49] and by integrating algorithms included in the OpenCV library and from the state of the art. This initial implementation was developed using a previous OpenCV version (OpenCV 2), therefore, an unifying and integrative task was necessary to adapt the available algorithms code to the newest version of the software as of June 2017 (OpenCV 3.2).

3.2 System architecture

The architecture of the proposed system is shown in Figure 3.1. It performs a frame level analysis over the input video sequence. For each stage of the system various algorithms performing each module task have been implemented. The algorithms to be executed are chosen by the user, according to the sequence under analysis, through the input parameters. As can be seen from the block diagram, the results provided by the system are written and saved in an output file by the event writer module. The system configuration and all the modules in between are described in detail in the following subsections.

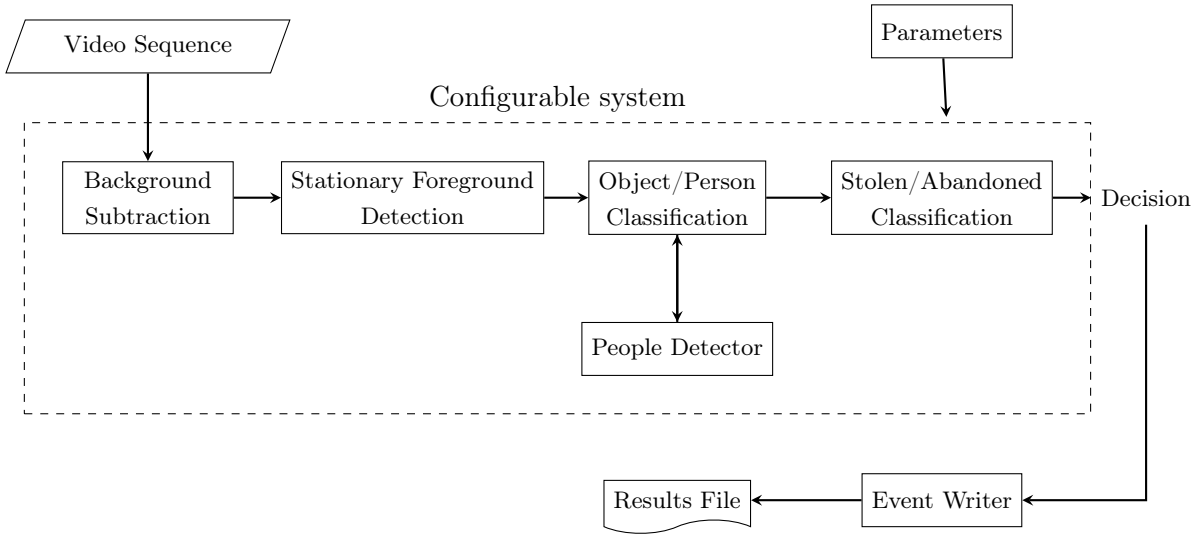


Figure 3.1: Proposed complete configurable system block diagram. For each block the system designer can select an algorithm from the available ones.

3.2.1 System configuration

Parameters that are required at the input of the system, as can be seen in Figure 3.1, are listed and described below.

- **Video file path**. This must be the complete path of the video file under analysis.
- **Algorithms identifiers (IDs)**. Each module requires an algorithm identifier that is represented by a integer number (e.g. 1, 2, 3...). The system requires four techniques identifiers for the following modules: background subtraction, stationary foreground detection, people detection and abandoned/stolen classification.
- **Output file name**. This must be the name of the output results file.
- **Output file folder path**. This must be the folder path of the results file.

These parameters are entered into the system through console inputs as in Figure 3.2.

3.2.2 Background subtraction module

Figure 3.3 shows the block diagram corresponding to the background subtraction module. This module receives as inputs each frame of the input sequence and the chosen background subtraction algorithm identifier. As outputs, it provides the background model, computed by the algorithm, and the foreground mask.

```
./Configurable_system 3 1 2 1 ./datasets/input_video.mpg videoName ./results/
```

Figure 3.2: Terminal command line example calling executable file followed by four algorithms identifiers, input video file path, output file name and output file folder path.

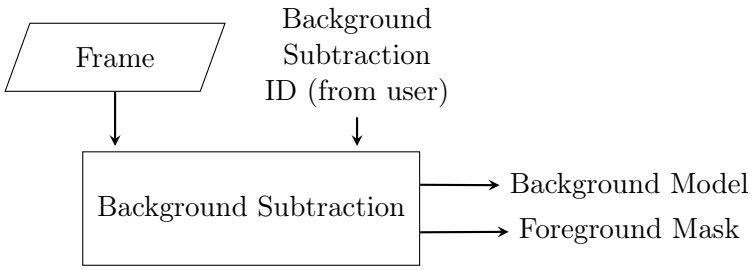


Figure 3.3: Background subtraction module block diagram

Five background subtraction techniques have been integrated in the background subtraction module: LOcal Binary Similarity segmenTER (LOBSTER) [50], Pixel-based Adaptive Word Consensus Segmenter (PAWCS) [51], Mixture of Gaussians (MOG2) [52], K-Nearest Neigbours (KNN) [53] and Independent Multimodal Background Subtraction (IMBS) [54]. Table 3.1 shows a summary of the robustness of each technique. The reason why LOBSTER, PAWCS and IMBS techniques have been considered is that their implementations were available in the Video Processing and Understanding Lab and they are recent state of the art options. In the case of MOG2 and KNN, the motivation was their availability in the OpenCV library. Default settings have been employed for all algorithms.

Technique	**ID**	**Training**	**Robustness**		
			Illumination changes, shadows and reflects	**Multimodal back-grounds**	**Occlusions**
LOBSTER [50]	1	No	Yes	Yes	Yes
PAWCS [51]	2	No	Yes	Yes	No
MOG2 [52]	3	No	Yes	No	Yes
KNN [53]	4	Yes	Yes	No	No
IMBS [54]	5	Yes	Yes	Yes	No

Table 3.1: Summary of integrated background subtraction algorithms characteristics and robustness

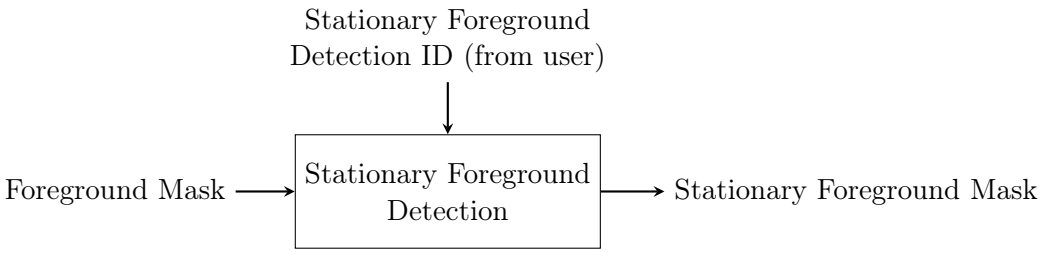


Figure 3.4: Stationary foreground detection block diagram

Technique	**ID**	**Robustness**			**Configurable parameters**
		Background model initialization	**Occlusions**	**Illumination changes**	
Subsampling [34]	1	No	Yes	No	Number of samplings
ACC [35]	2	No	Yes	No	Detection threshold

Table 3.2: Summary of integrated stationary foreground detection algorithms robustness and parameters .

3.2.3 Stationary foreground detection module

The block diagram for the stationary foreground detection module is shown in Figure 3.4. As it can be seen, this module receives as inputs the foreground mask, previously computed by the background subtraction module, and the chosen stationary foreground detection algorithm identifier. As output it provides the stationary foreground binary mask indicating which areas of the foreground remains still. A key parameter of this module implementation is the "time to static" parameter. This parameter determines how long an object has to remain still in order to be consider as static. Throughout this work, this parameter has been set to 10 seconds.

Two stationary foreground detection algorithms have been integrated in this module, both of them are explained in Section 2.3.2: foreground mask subsampling (Subsampling) and accumulation of foreground masks (ACC), which procedure is shown in Figure 3.5. Table 3.2 summarises their robustness and configurable parameters for each algorithm. These techniques have been chosen because they are basic state of the art techniques with low computational cost and provide reasonable results.

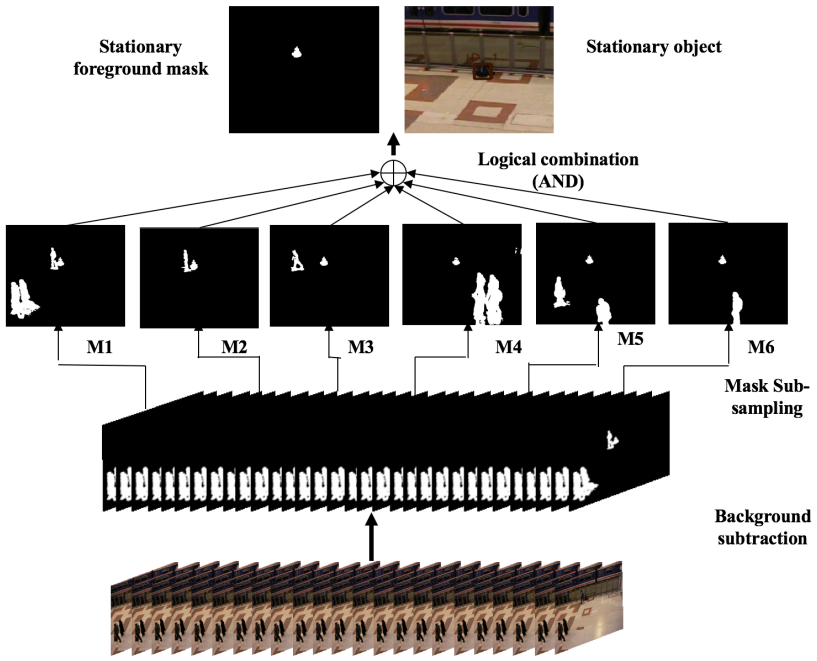


Figure 3.5: Foreground mask subsampling procedure [33].

3.2.4 Object/people classification and people detection modules

Figure 3.6 shows the block diagram of both object/people classification and people detection modules. Firstly, from the stationary foreground mask, provided by the stationary foreground detection module, a list of the stationary areas in the scene in generated.

The people detector module receives as inputs the chosen algorithm identifier and the stationary regions list. For each region, it provides at the output a detection score, which determines the likelihood of such region of being a person. Five different people detection algorithms have been integrated in the developed system, all of them are based on appearance and their characteristics are summarised in Table 3.3.

The classification module requires at the input the stationary foreground mask, and the previously mentioned score provided by the people detector. Depending on how large the score is, the region is classified as an object or as a person. It is assumed that everything not classified as a person is considered as an object. As a result of this classification the module provides at the output a list of all the regions classified as objects, i.e. a list of objects in the scene, that is what we are interested in.

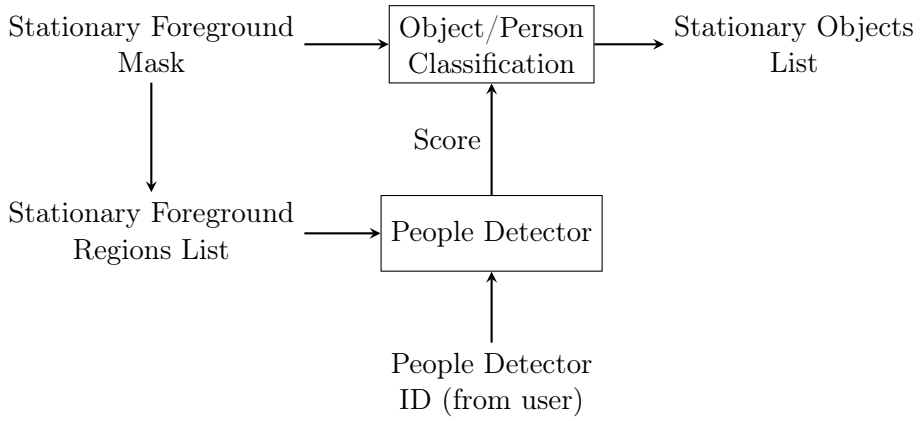


Figure 3.6: Object/People classification and people detection module block diagram

Technique	**ID**	**Type**	**Training model**
HOG	1	Holistic classifier	SVM classifier with person template
DPM	2	Fusion classifier	Latent SVM w/ global and parts person templates
Haar Upper Body	3	Cascade classifier	Human upper body template
Haar Frontal Face	4	Cascade classifier	Frontal human face template
Haar Full Body	5	Cascade classifier	Human body template

Table 3.3: Summary of integrated people detection techniques characteristics.

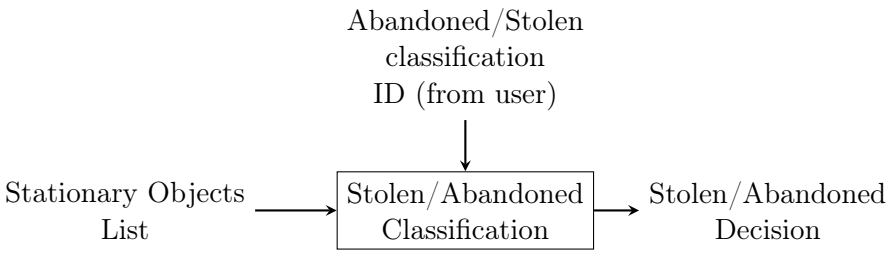


Figure 3.7: Abandoned/Stolen classification module block diagram

Technique	**ID**	**Feature**
High Gradient	1	Edges
Color Histogram	2	Color

Table 3.4: Summary of the integrated classification algorithms identifiers and features.

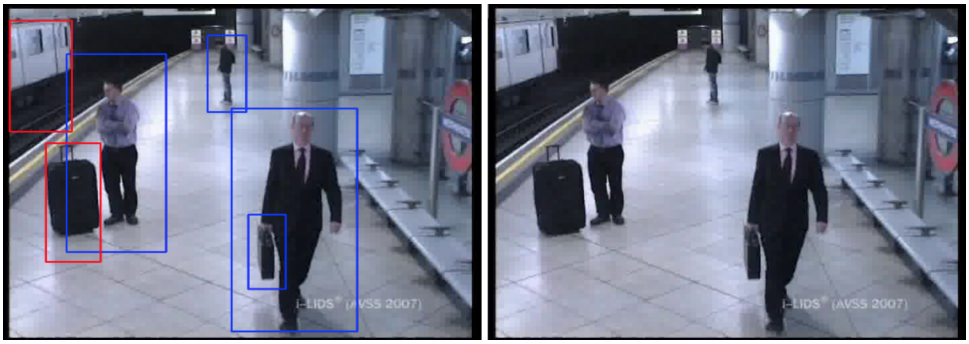
3.2.5 Abandoned/Stolen classification module

Figure 3.7 shows the block diagram of the module in charge of the object classification as abandoned or stolen. This module receives as inputs the stationary objects list and the chosen classifier algorithm identifier. At the output, for each object in the list, it provides the decision taken for each one.

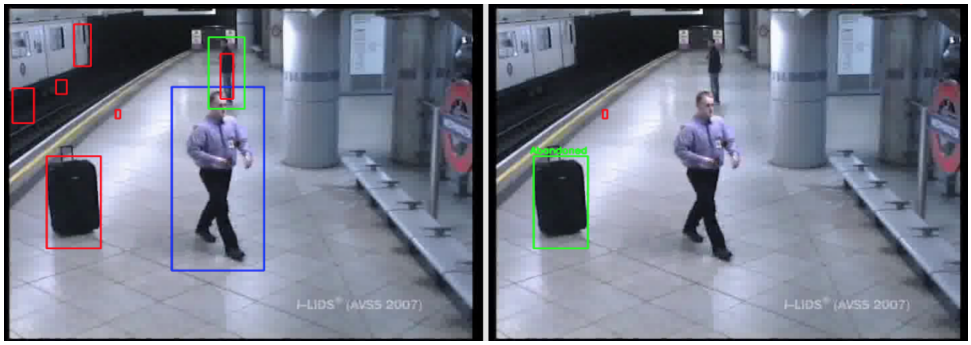
An object that initially was not in the scene and subsequently is unattended by a person is considered as an abandoned or unattended object. The fact that a person unattends an object means that he has to leave it somewhere in the scene and has to move away from it, which means that even if an object is remaining still in the scene if a person is detected close to it the object will not be considered as unattended. For instance, a suitcase on a platform will not be detected as abandoned until the person who left if off moves away from it. In this work the minimum distance a person has to walk away from the object has been considered as twice the width thereof. Examples are shown in Figure 3.8.

An object is considered as stolen if it has been present in the scene from the beginning, being part of the scene background, and at any time a person removes it. It is understood that an object cannot be stolen or abandoned by itself without human interaction. This situation may occur in a museum or exhibit space.

Two classification algorithms, listed in Table 3.4, have been integrated in the system. Both High Gradient and Color Histogram algorithms are explained before in Section 2.5.2.



(a) Attended object scenario.



(b) Unattended object scenario.

Figure 3.8: Examples of attended and unattended luggage scenarios. In both sub-figures left image shows the detections made by the system (red stands for candidates, blue for detected people). Right image is showing the final detections with their classification. In Sub-Figure 3.8a the owner of the suitcase is standing close to it thus it is not detected as abandoned, however, in Sub-Figure 3.8b the owner is walking away from the suitcase, and he is far enough to the suitcase to be detected to abandoned.

Figure 3.9: Event writer module block diagram

```
<file id="0" name="Information">
<attribute name="SOURCETYPE">
<data:lvalue value="SEQUENCE"/>
</attribute>
<attribute name="NUMFRAMES">
<data:dvalue value="3330" />
</attribute>
<attribute name="FRAMERATE">
<data:fvalue value="25.0" />
</attribute>
<attribute name="H-FRAME-SIZE" >
<data:dvalue value="320" />
</attribute>
<attribute name="V-FRAME-SIZE" >
<data:dvalue value="256" />
</attribute>
</file>
<object framespan="1107:2814" id="0" name="StolenObject">
<attribute name="BoundingBox">
<data:bbox height="12" width="13" x="124" y="86"/>
</attribute>
</object>
```

Figure 3.10: Extract from an output file

3.2.6 Event writer module

The last module of the system block diagram is displayed in Figure 3.9. The event writer module aims to note down the abandoned and stolen objects detected in a XML file. Having the results noted is essential for posterior system evaluation.

This module receives the detections achieved for each frame, it processes the information frame by frame and finally write down them in the output file. Figure 3.10 shows a fragment of a results file where a stolen object detection has been annotated. For each noted event its initial and final frames and its position within the scene are also saved.

Internal procedure of the module is shown visually with a block module in Figure 3.11. Stationary objects, i.e. events, of each frame are entered into *Detect New Events* module, this module simply creates a list called *New Events,* containing all events detected in every frame. *Check New Events* module is in charge of checking if an event detected in a frame already exists or it is the first time it is detected. This is done by comparing them with events in *Active Event*s, which contains ongoing events along the sequence. If a detected event is detected for the first time it is added

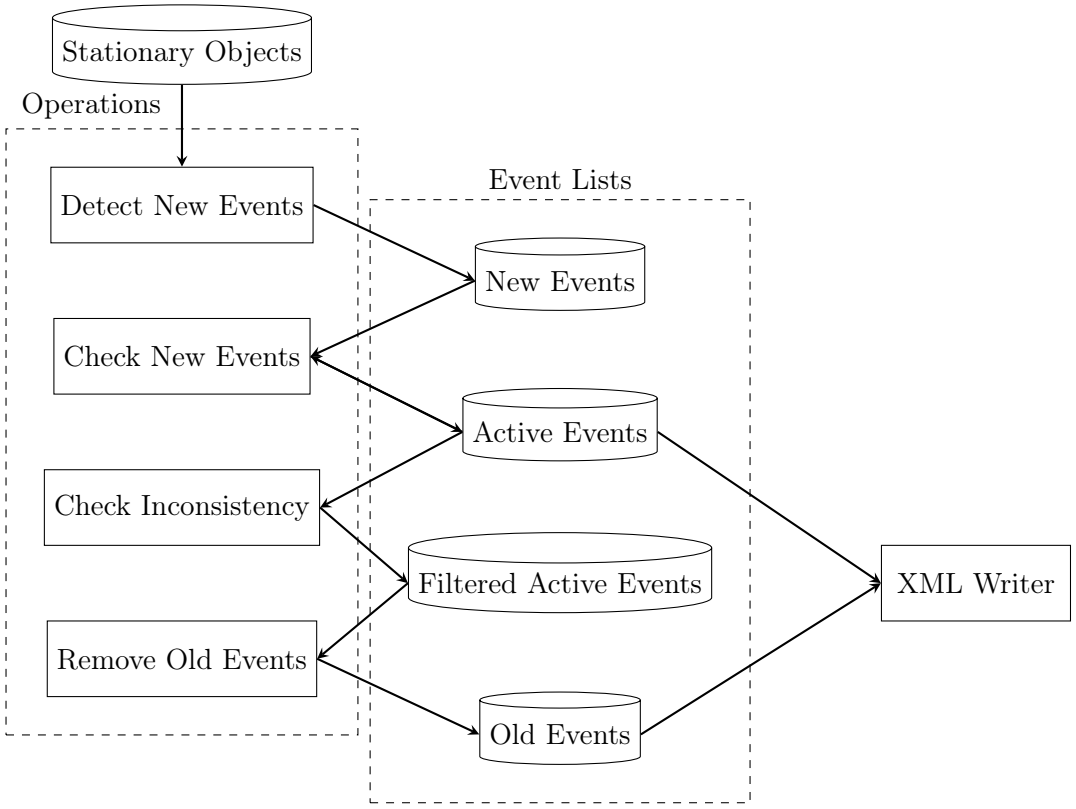


Figure 3.11: Event writer module operation.

to *Active Events* list, on the contrary it is updated. *Check Inconsistency* module filters *Active Events* list by removing duplicated events whether spatial or temporally. *Remove Old Events* stage checks *Filtered Active Events* and moves old events to *Old Events* list, once an event is moved to *Old Events* it is removed from *Filtered Active Events*. An event is considered as "old" when it has not been detected for a while or its lifespan exceeds a preset number. Finally, at the end of the sequence all old and active events are written to the results file.

3.3 Graphical User Interface

A graphical user interface (GUI) has been also developed. The aims of this interface are to act as a demonstrator, which allows the user to check the system functionality in a visual way, and to manually set the system algorithms and parameters.

The development environment used for the interface creation has been Qt Creator (version 4.2.1) based on Qt 5.8.0. Qt Designer is the design tool incorporated in Qt Creator software, an overview is shown in Figure 3.12. In Section 3.3.1 all the

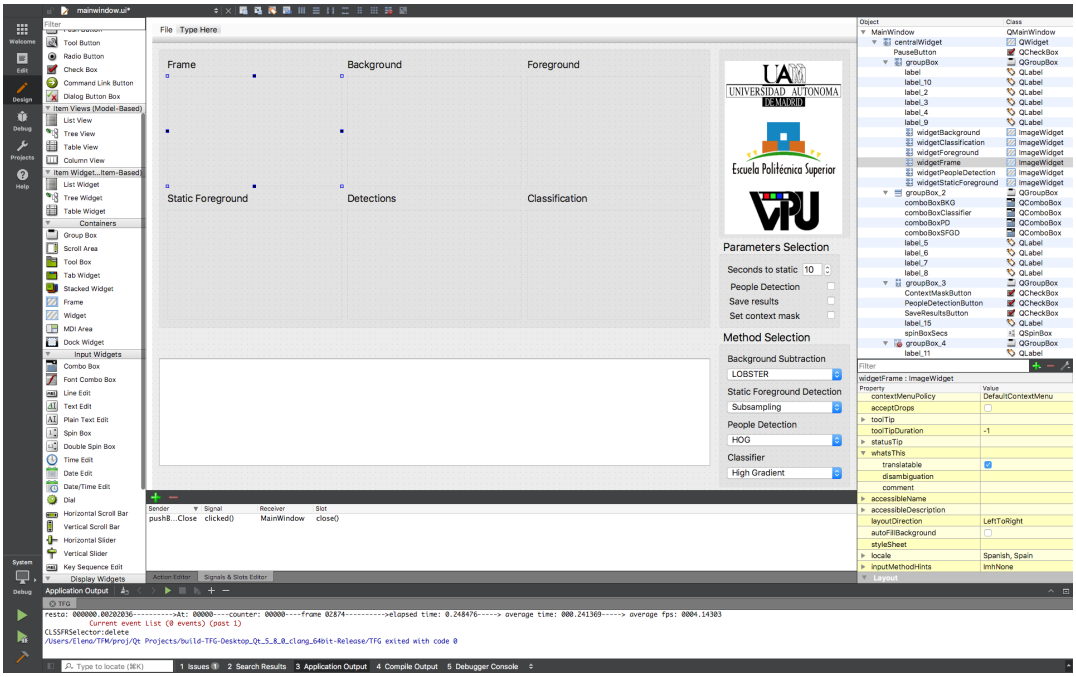


Figure 3.12: QT design environment.

functionality of the interface is explained.

3.3.1 GUI functionality

Observing Figure 3.13, where an general overview of the interface is shown, it can be seen that it is composed of several elements, described in detail below.

- **Menu bar**. This bar, shown in Figure 3.14, is placed on the top of the screen and the "file" option allows the user to select the desired vile file to process from the file explorer. Several formats (*.mpg, *.avi, *.mov, *.mp4, ...) are supported.

- **Display area**. This area, shown in Figure 3.15, allows the user to visualize the results of the system after each stage thereof. On top row from left to right, the current frame, the computed background and the foreground mask are displayed. On bottom row, the static foreground mask is displayed first, the people and object detections are shown in blue and red color respectively in the second display and finally the stolen/abandoned object classification.

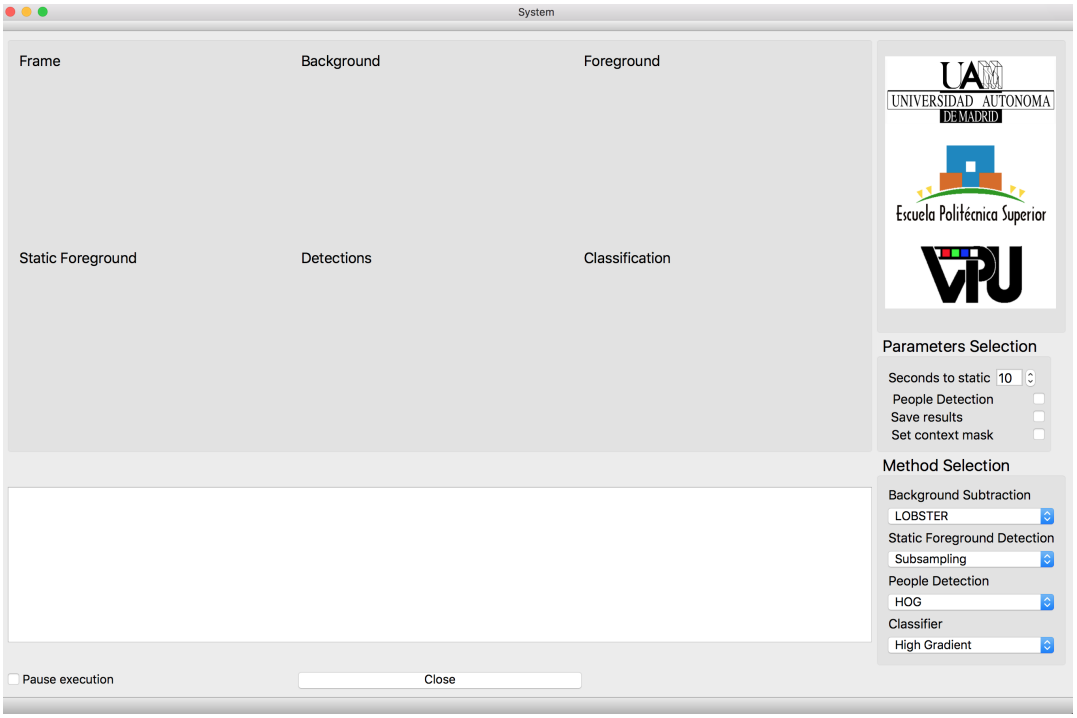


Figure 3.13: Graphical User Interface overview.

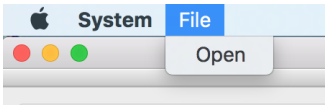


Figure 3.14: Menu bar of the GUI.

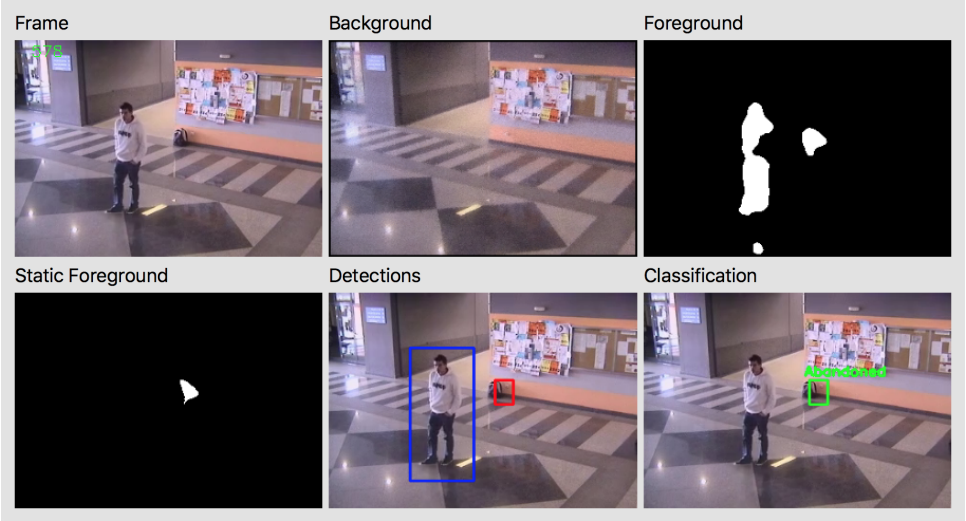


Figure 3.15: Display area of the GUI.

- **Algorithm selection**. As it is shown in Figure 3.16, user can choose a different algorithm for each module of the system by selecting it in a drop-down menu.

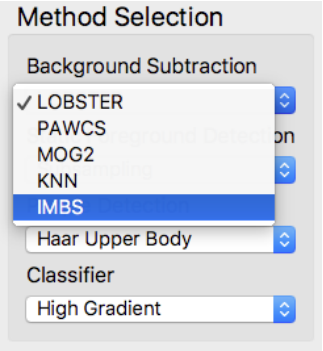


Figure 3.16: Method selection of the GUI.

- **Parameters selection**. User can choose or modify the following parameters of the system, as shown in Figure
  - ✧ *Seconds to static*. This option allows the user to modify the number of seconds until an object is considered as stationary. It is set to 10 seconds by default.
  - ✧ *People detection*. If this checkbox is activated, the people detection is running throughout the full sequence, otherwise it only will be run when something static is detected.
  - ✧ *Save results*. If it is checked a XML file with the detections found along the sequence will be created and saved.
  - ✧ *Set context mask*. This option allows the user to select an area of the frame where the detections will be ignored (non-interest area).

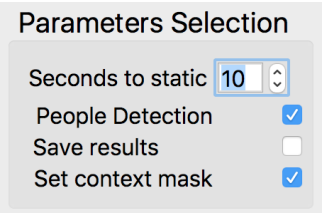


Figure 3.17: Parameters selection of the GUI.

- **Dialog box**. As can be seen in Figure 3.18, results are displayed in real time in this dialog box. Also, pause and close the execution options have been implemented.

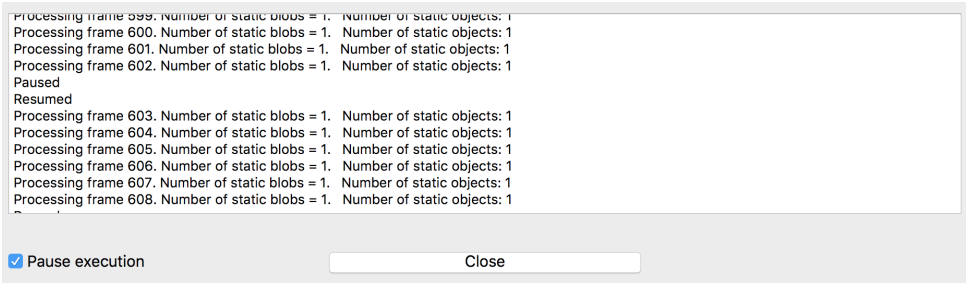


Figure 3.18: Dialog box of the GUI.

Chapter 4

Proposed system

Aiming to avoid background subtraction algorithms dependance, several works have been recently proposed in the literature, such as [55], make use of dependence across color channels for abandoned object detections, or [56], consider spatio-temporal changes for detection.

Based on [56], a new system is proposed in this chapter to improve the performance of the system described in the previous chapter and to deal with long-term video sequences challenges, mentioned in Section 1.1. Due to the fact that this processing strategy is different than the one applied until now, it was not possible to integrate it in the configurable abandoned and stolen objects detection system, consequently a new system has been necessarily created.

The general overview of the new proposed system and the implemented algorithms are described hereunder.

4.1 Overview

The proposed new abandoned and stolen objects detection system main architecture is shown in Figure 4.1. It is primarily composed of five main modules: *Spatio-Temporal Change Detection* [56], *LaBGen* [57], *People Detector, Sudden Illumination Changes Detector* and *Classifier*.

Comparing this block diagram with the presented in the previous chapter, in Figure 3.1, one can observe that the main concept is now simpler than before: to detect abandoned and stolen objects by means of stability changes over the frames of the video sequences. When an object is abandoned or stolen it is easy to see that something in the scene undergoes visual changes, a change in the frame occurs. The aim of the *Spatio-Temporal Change Detection* module is, precisely, to detect changes in

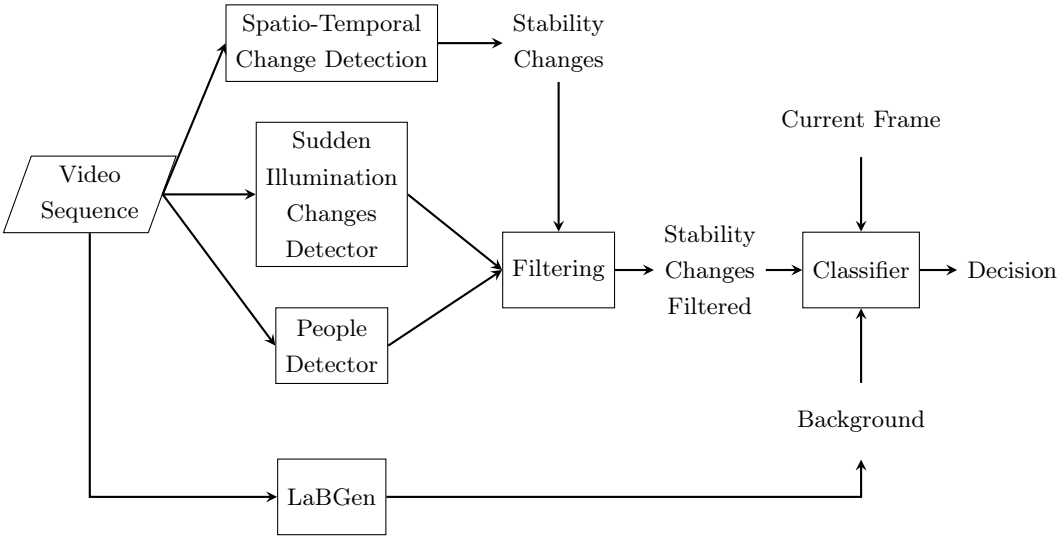


Figure 4.1: Complete configurable system block diagram

the frame stability along the sequence. *LaBGen* module performs a stationary background generation method able to compute a stationary background image dealing with traditional background subtraction algorithms challenges. *People Detector* and *Sudden Illumination Changes Detector* modules are used to filter the detections when they occur due to people presence or due to a sudden illumination changes, for instance, when light is suddenly switched on or off. Making use of the filtered stability changes detected and the background computed by *LaBGen* module, an abandoned/stolen classifier determines if the change in the stability frame was due to an abandon or a stealing.

4.2 LaBGen module

4.2.1 Introduction to LaBGen method

LabGen module of the proposed system performs LaBGen method, that was initially proposed in [57]. It obtains the best performance among the stationary background generation methods submitted to the Scene Background Modeling and Initialization (SBMI) workshop organized in 2015 by Maddalena and Bouwmans [58]. It also achieved first rank during the IEEE Scene Background Modeling Contest (SBMC) organised in 2016.

LabGen algorithm is able to generate a stationary background image even when the background is never fully visible, this can be due to people walking or cars moving. The algorithm, in a nutshell, is based on combining the principles of a pixel-wise

Figure 4.2: Left two images show two situations where the background is occluded and right image is the stationary background image LabGen can produce from a sequence presenting situations like the ones on the left. Image from [59].

temporal median filter with a mechanism to select patches containing lowest quantities of motion. An example of LabGen background estimation is shown in Figure 4.2.

4.2.2 Method description

This section will provide a brief overview of LaBGen method. It comprises five steps:

1. An augmentation step increases the length of the input video sequence (in case of short videos). Parameter P controls the length of the augmented video sequence.

2. A frame per frame motion detection step determines which pixels belong to the background. The parameter A identifies the used motion detection algorithm.

3. Quantity of motion estimation is done locally based on the motion detection, inside of spatial areas whose size is dependent on the parameter N.

4. Selection of a subset of patches with the least motion, based on the resulting quantities of motion, for each spatial area. Parameter S determines the subset size.

5. Generation of the stationary background image B by applying the temporal median filter on the subsets of selected patches.

4.2.3 Integration

In the developed system LaBGen-P has been integrated. LaBGen-P is based on LaBGen and instead of computing a quantity of motion for a given patch, LaBGen-P computes quantities of motion per pixel by taking into account the motion in the spatial neighborhood of each considered pixel [60]. Selecting pixels instead of patches

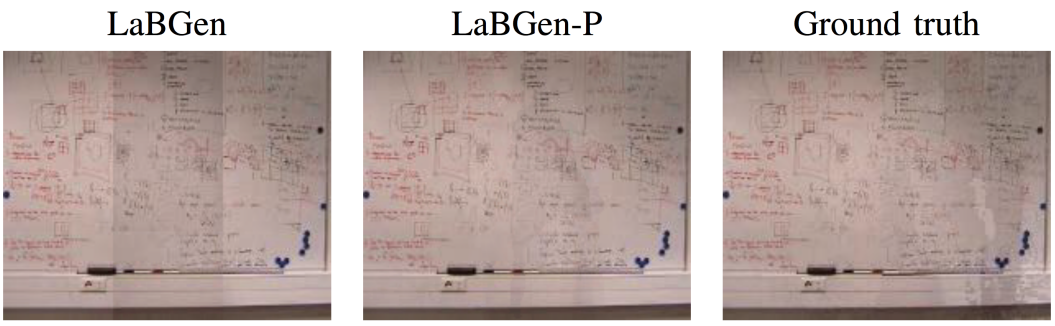


Figure 4.3: Background estimation comparison between LaBGen and LaBGen-P. Image taken from [60].

has the advantage of avoiding spatial discontinuities and providing a more consistent background estimation. Figure 4.3 illustrates this issue.

Following the authors recommendations, parameters have been set at their default values:

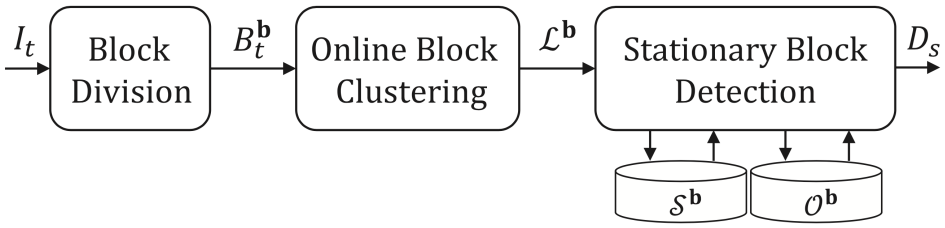
- $A = Frame\ Difference$. Frame difference is a simple motion detection with low computational cost and giving best mean performance [57] on SBI 2016 dataset [58].

- $N = 3$. Using $3x3$ pixels patches.

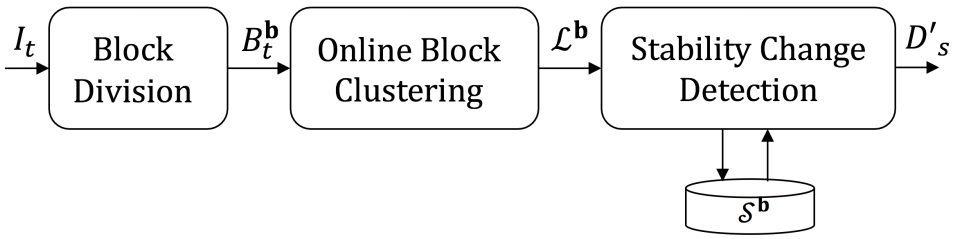
- $S = 19$. For each NxN patch, a temporal buffer of 19 patches with least motion it taken.

LaBGen-P has been integrated as follows. The *LaBGen* module receives as input a number of frames and computes a stationary background that will be used by the classifier module in order to classify between abandoned and stolen. This module performs at the same time and separately from the previous module.

The frequency for each background computation is a crucial issue, because keeping in mind that the background image is computed by applying a median filter and the time an object needs to remain still in order to consider it as abandoned or stolen, it has to be large enough to not to include the object we want to detect. For this reason LaBGen background estimation is carried out every $((2 \cdot time\_to\_static \cdot fps) + 10)$ frames, i.e. twice the number of frames the object has to remain still plus a little offset (10 frames). That way we ensure that the change in the scene we want to classify is not present in the background reference image.



(a) Original block diagram of the method proposed in [56].



(b) Integrated block diagram module in the proposed system.

Figure 4.4: (a) Original stationary object detection approach based on spatio-temporal change detection [56] and (b) integrated approach block diagrams.

4.3 Spatio-Temporal Change Detection module

4.3.1 Introduction

Spatio-temporal change detection module, is based on [56]. This work propose a long-term stationary object detection based on spatio-temporal change detection. This approach has several advantages, as it does not make use of background subtraction algorithms to perform the stationary objects detection, it is not constrained to their limitations. It is also robust to illumination changes and it can quickly be adapted to scene variations. In addition, unlike most of the state of the art approaches, where many parameters and thresholds are needed, few parameters are required.

The reasons why this approach has been integrated in the proposed system are the mentioned advantages, the availability of the source code and the good results obtained when it was validated for short-term and long-term scenarios in [56].

4.3.2 Description

This section will provide a brief description of the method proposed in [56], that is the baseline of the spatio-temporal change detection module in the proposed system. Its stages are shown in a block diagram in Figure 4.4a.

At the beginning, each frame I_t (where t stands for each time instant) is divided

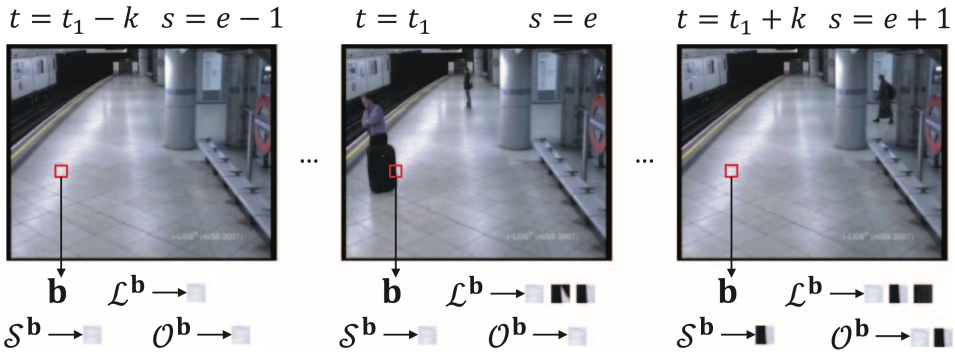


Figure 4.5: Temporal analysis for a block located in **b** where a suitcase appears and is removed after. Figure from [56].

into NxN non-overlapping blocks $B_t^{\mathbf{b}}$ (**b** denotes location) in the *block division* stage. Next module, *online block clustering*, is in charge of modeling the temporal scene evolution by grouping blocks that are similar into the same clustering, updating cluster partitions $\mathcal{L}^{\mathbf{b}}$. Each block $B_t^{\mathbf{b}}$ at the input is assigned to a cluster in $\mathcal{L}^{\mathbf{b}}$ or a new one is created if required. Lastly, *stationary blob detection* stage analyses scene stability to identify stationarity at regular sampling instants. For this purpose, last stable clusters $S^{\mathbf{b}}$, old stable clusters $\mathcal{O}^{\mathbf{b}}$ and alarm time T are used. This module provides at the output a results image D_s where s stands for the sampling instant. At each sampling instant, the most stable cluster from $\mathcal{L}^{\mathbf{b}}$, denoted by $C_s^{\mathbf{b}}$, is obtained, and in order to check if a stability change has occurred, it is compared with the last most stable cluster $S^{\mathbf{b}}$. In case they are different, it is also compared with $\mathcal{O}^{\mathbf{b}}$ in order to check if it is an old stable cluster. In case a stability change occurs and it is not similar to any old stable cluster and it exceeds the alarm time, a stationary detection will appear in the output image D_s.

Figure 4.5 shows an example of the temporal analysis for a block location **b** (red square). In this situation a suitcase, that was not initially in the scene remains still awhile and is removed later. $\mathcal{L}^{\mathbf{b}}$ is keeping clusters in **b**, $S^{\mathbf{b}}$ is keeping the last stable one and $\mathcal{O}^{\mathbf{b}}$ the old stable clusters in that location.

4.3.3 Integration

As seen above, in Section 4.2, *LaBGen* module is computing a stationary background scene estimation every certain amount of frames. For this reason, keeping old stable clusters $\mathcal{O}^{\mathbf{b}}$ has no longer sense. Block diagram of *spatio-temporal change detection* module is shown in Figure 4.4b. Small differences regarding original block diagram in Sub-Figure 4.4a can be appreciated. Now, the most stable cluster from $\mathcal{L}^{\mathbf{b}}$ is only

compared with the last stable cluster $S^{\mathbf{b}}$ in order to detect if there has been a change in the stability, and once the alarm time has been exceeded, the new output image D'_s will return a stability change in $\mathbf{b}$. The posterior classifier will be responsible for the stolen/abandoned classification taking into account the background and current frame information.

4.4 Filtering

The provided code of this implementation [61] included two additional modules for filtering that have been kept. A *People Detector* module performing Aggregated Channel Features (ACF) [44], and a *Sudden Illumination Changes Detector* module.

People Detector module provides a mask containing the people detected in the scene. This is helpful for avoiding possible false detections caused by people standing without moving.

Sudden Illumination Changes Detector module allows to avoid false detections caused, for instance, when light is suddenly switched on or off. This module is based on detecting changes in the entropy of the scene.

4.5 Abandoned/Stolen Classifier

In order to classify the stability changes that have been detected in the scene into abandoned or stolen, the same two algorithms (Color Histogram and High Gradient) integrated in the configurable system have been integrated in this one. Their explanation can be found in previous Section 2.5.2.

Chapter 5

Evaluation methodology

This chapter provides the evaluation methodology description. The tested sequences classification will be described as well as the employed evaluation metrics and the software that have been used for evaluation and ground-truth annotation.

5.1 Dataset classification

To design a protocol evaluation the following datasets have been classified into three difficulty categories (easy, medium and hard) depending on its robustness against several challenges, introduced in Section 1.1.

- **AVSS AB 2007.** Three sequences of abandoned baggage scenario in a metro station platform. Sequences are $720x576$ pixels size, they last approximately 3.5 minutes each and they are available at www.eecs.qmul.ac.uk/~andrea/avss2007_d.html.

- **CAVIAR**. Four sequences of abandoned and picked up objects recorded from above of the scene. Sequences are $384x288$ pixels size, they last approximately 1 minute each and they are available at www.homepages.inf.ed.ac.uk/rbf/CAVIAR/.

- **PETS 2006**. Nine sequences of abandoned objects scenario recorded in a railway station from three different cameras with different points of view. Sequences are $720x576$ pixels size, they last almost 2 minutes each and they are available at www.cvg.reading.ac.uk/PETS2006/data.html.

- **VISOR** (Stopped Vehicle). Four sequences for stopped vehicles detection. Sequences are $320x256$ pixels size, they last from 1 to 3 minutes and they

are available at www.openvisor.org/video_videosInCategory.asp?idcategory=12.

An example frame of each database is shown in Figure 5.1.

In total, 20 video sequences have been viewed and classified. The carried out classification is shown in Table 5.1. The classification has been done manually by taking into account several difficulties the video sequence may contain. These challenges are illumination changes or shadows moving, movements in the stationary background, occlusions and small objects due to remoteness. Depending on the presence of these challenges in the sequences ($-$ stands for low, $+$ for medium, $++$ for high and $+++$ for very high) they have been classified into *Easy* (E), *Medium* (M) and *Hard* (H) categories.

In order to evaluate long-term, two more video sequences have been considered:

- **AVSS AB 2007**. One video sequence of abandoned and picked up baggage scenario in a metro station platform. This sequence lasts 21 minutes and it is $720x576$ pixels size. It is available at www.eecs.qmul.ac.uk/~andrea/avss2007_d.html.

- **AVSS PV 2007**. One video sequence of parked vehicles scenario. It lasts 18 minutes and it is $720x576$ pixels size. It can be found at www.eecs.qmul.ac.uk/~andrea/avss2007_d.html.

Two example frames of these datasets are shown in Figure 5.2. It is important to note the lack of available long term surveillance video sequences.

5.2 Evaluation metrics

In order to evaluate the performance of the developed system, the obtained results have been compared with their ground-truth using ViPER-PE tool of ViPER software (Video Performance Evaluation Resource www.viper-toolkit.sourceforge.net/). ViPER-PE is a command line performance evaluation tool that allows the user to select from multiple metrics to compare a result data set with ground truth data.

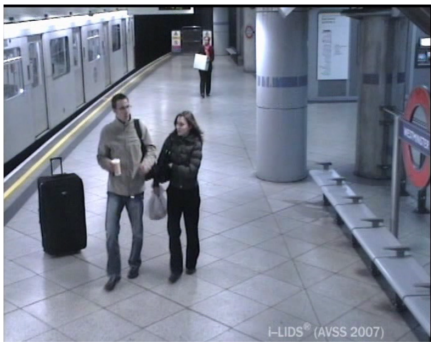
The three different major types of analysis in ViPER-PE are object, frame wise and tracking analysis. Object analysis is considered in the proposed evaluation protocol and it determines which targets (truth) and candidates (results) are close together, and then generates the precision and recall of the number of candidates matching targets.

(a) PETS 2006 Camera 1

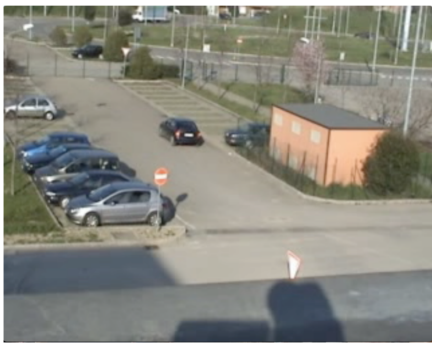
(b) PETS 2006 Camera 3

(c) PETS 2006 Camera 4

(d) CAVIAR



(e) AVSS AB 2007



(f) VISOR

Figure 5.1: Example frames of short-term evaluated datasets.

	Challenges						
Sequence	Illumination / Shadows	BKG movement	Occlusions	Far objects	Complexity	#AB	#SO
VISOR_00	-	-	-	-	E	2	1
VISOR_01	+	-	-	-	E	1	1
CAVIAR_LB_PU	+	+	-	-	E	1	1
AVSS2007_AB_E	+	+	-	-	E	1	1
PETS06_S1_C3	+	+	-	-	E	1	0
PETS06_S5_C3	+	+	-	-	E	1	0
VISOR_02	++	+	-	-	M	2	2
CAVIAR_LeftBag	+	+	-	-	M	1	1
CAVIAR_LeftBox	+	+	-	-	M	1	0
AVSS2007_AB_M	+	++	+	+	M	4	4
PETS06_S4_C3	+	+	-	-	M	1	0
PETS06_S4_C4	+	++	+++	+	M	1	0
PETS06_S1_C1	+	+++	-	-	M	1	0
VISOR_03	++	++	-	-	H	1	1
CAVIAR_LB_AC	+	++	-	++	H	1	1
AVSS2007_AB_H	+	++	++	+	H	1	0
PETS06_S1_C4	+	+++	-	+	H	2	2
PETS06_S5_C1	+	+++	-	-	H	3	2
PETS06_S5_C4	+	+++	-	++	H	1	0
PETS06_S4_C1	++	+++	+++	-	H	1	0

Table 5.1: Complexity classification of the short-term sequences under analysis regarding different challenges into three complexities: E = easy, M = medium and H = hard. (- low presence, + medium presence , ++ high presence, +++ very high presence). #AB stands for number of abandoned objects and #SO for number of stolen objects.

(a) AVSS AB 2007 (b) AVSS AB 2007

Figure 5.2: Example frames of long-term evaluated datasets.

```
#BEGIN_OBJECT_EVALUATION

OBJECT AbandonedObject [dice 0.99]
    BoundingBox : [dice 0.99]

OBJECT StolenObject [dice 0.99]
    BoundingBox : [dice 0.99]

#END_OBJECT_EVALUATION
```

Figure 5.3: Example of Evaluation Parameter File for object evaluation.

The selected metric is *dice* coefficient, that is defined as twice the shared area, divided by the sum of the two areas:

$$dice\ coeff. = 1 - \frac{2|A \cap B|}{|A| + |B|}$$

There are three possible types of evaluation: object evaluation, frame wise evaluation and tracking evaluation. Object evaluation is the one that has been considered. Descriptors (e.g. abandoned and stolen objects) and attributes to evaluate (e.g. bounding box, frame span) are defined in an Evaluation Parameter File (EPF). This file tells the system how to perform an evaluation of candidate data against a target, the employed EPF file is shown in Figure 5.3 as an example.

As can be seen in the EPF file, for each detected event two attributes are considered: its location, in terms of a bounding box, and its lifespan, in terms of first and finish frame. Following the EPF file and event in considered as detect as follows:

$$Event\_Detected = \begin{cases} True & if\ temporal\_dice\ coeff. < \sigma\ and \\ & spatial\_dice\_coeff. < \tau \\ False & otherwise \end{cases}$$

where σ and τ have been set to 0.99. To evaluate the obtained results of the proposed systems the following measures have been considered:

- **Precision** (P), also also called positive predictive value (PPV), indicates how many of the detections made are relevant, i.e. it states the rate between the correct detections and the total number of detections made:

$$Precision = \frac{TruePositives}{(TruePositives + FalsePositives)} \tag{5.1}$$

- **Recall** (R), also known as sensitivity, indicates how many relevant detections have been made, i.e. it states the rate between the correct detections and the number of ideal correct detections:

$$Recall = \frac{TruePositives}{(TruePositivies + FalseNegatives)} \tag{5.2}$$

- **F-Score** (F), combines in a unique value both precision and recall measures in a well-considered way:

$$F - score = \frac{(2 \cdot Precision \cdot Recall)}{(Precision + Recall)} \tag{5.3}$$

5.3 Ground-truth

Ground-truth files of short-term videos sequences have been provided, however the two long-term sequences ground-truth files have been manually created with ViPER-GT tool. ViPER-GT is a Java graphical user interface, designed to allow frame-by-frame markup of video metadata stored in the Viper format. It is shown in Figure 5.4.

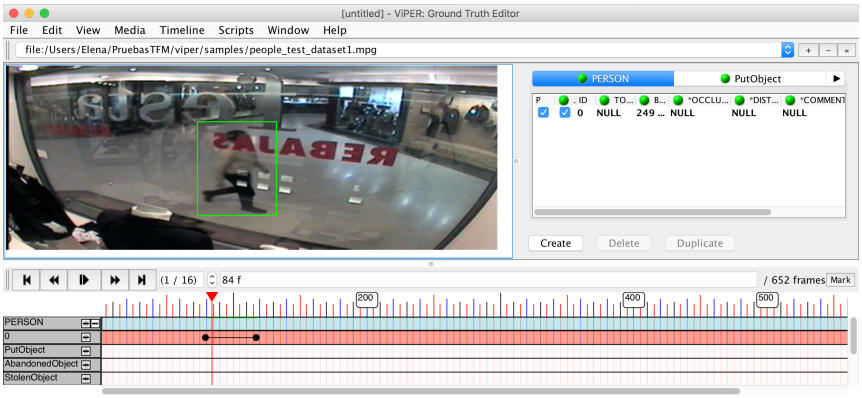


Figure 5.4: ViPER-GT screenshot.

It is important to remark that all ground truth files employs "time to static" parameter = 10 seconds. This means, for instance, that an unattended object is considered as abandoned after it has remained still 10 seconds.

Chapter 6

Results

This chapter provides an analysis of the obtained results for both systems that have been implemented. Both of them have been tested over the same 22 video sequences (20 short-term sequences and 2 long-term sequences). The obtained results are studied and graphical examples of both systems performances are shown. In addition, a computational cost comparison has been made.

6.1 Configurable system accuracy

6.1.1 Configurations

In order to avoid the arduous task of evaluating all possible combinations combining all the algorithms implemented in each module of the configurable system, efforts have been made to select the more relevant combinations.

As seen in Chapter 3, four modules of the system require algorithms identifiers: background subtraction, stationary foreground detection, people detection and stolen/abandoned classifier. As every modules is completely reliant on results of the previous one, it is reasonable to try to achieve the best possible results at the first stage, background subtraction. For this reason, the three more promising implemented algorithms have been tested.

Regarding next stage, stationary foreground detection, subsampling and accumulation approaches have been integrated. Taking into account results of [49], where subsampling approach provided slightly better results than accumulation approach, it has been decided to evaluate subsampling approach.

As no comparison between Color Histogram and High Gradient approaches was available, concerning abandoned/stolen classification module, both have been tested combined with the best background subtraction performance.

	Background Subtraction	Static Foreground Detection	Classifier	People Detection
C1	PAWCS	Subsampling	High Gradient	HOG
C2	PAWCS	Subsampling	Color Hist.	HOG
C3	MOG2	Subsampling	Color Hist.	HOG
C4	IMBS	Subsampling	Color Hist.	HOG

Table 6.1: Algorithms combinations, i.e. configurations, that have been tested in the configurable system.

	E (Easy)			M (Medium)			H (Hard)		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
C1	81.67	100	89.91	25.14	50	33.46	6.71	28.57	10.87
C2	73.33	88.83	80.34	26.86	64.29	37.89	9.57	41.43	15.55
C3	8.50	88.83	15.52	3.57	50	6.67	1.57	45.71	3.04
C4	12.167	61	20.29	4.86	21.43	7.912	1.57	14.29	2.83

Table 6.2: Configurable system results for short-term video sequences with parameter $timeToStatic = 10$ seconds. Best performances marked in red.

For people detection module, due to the fact that in the sequences under analysis people appears under multiple points of view, only full-body detectors have been consider for this evaluation. Finally HOG detector was chosen for evaluation because it performs reasonable, although its results are slightly below DPM results [38], but it computational cost is lower.

The four configurations (C1, C2, C3 and C4) that have been tested over all the selected sequences are described in Table 6.1.

6.1.2 Short-term

In Table 6.2 average results, in terms of precision (P), recall (R) and F-score (F), of short-term video sequences evaluated are shown. The results are displayed according to the complexity (easy, medium or hard) and the four configurations. Best performance for each complexity is marked in red.

In the light of the results obtained it is noticeable that PAWCS algorithm for background subtraction outperforms the others. This results are illustrated in Figure 6.1 where same event of the same sequence is evaluated under MOG2, IMBS and PAWCS background subtraction algorithms. The three tests have been made with Subsampling approach for static foreground detection, High Gradient algorithm at

the classifier and HOG people detector.

The event under analysis is an abandoned bag in a train station scenario. Looking at top image in Figure 6.1, where MOG2 background subtraction is running, it can be seen that a lot of false positives are detected. The reason for this detections is that the algorithm is not modeling changes in the background, such as people moving in the upper part of the image, and this continuous movement of people in the same area is generating continuous foreground which means that static regions are generated, as can be seen where the static foreground is displayed. In short, MOG2 algorithm is able to detect the abandoned bag correctly, but it also generates many false detections, that is why its precision, in Table 6.2, is really low.

Looking at middle set of images in Figure 6.1, where IMBS background subtraction is running, one can observe that the abandoned bag is not detected. The reason IMBS is not detecting the bag is that default settings for this algorithm make that changes in the scene are quickly absorbed into the background, in the background display is shown that the bag has been completely added to the background, thus it cannot be detected as a stationary object.

Finally, looking at the set of images at the bottom, where PAWCS is running, it can be seen that the abandoned bag is perfectly detected. PAWCS is able to correctly detect and model movements in the background, such as people in the chairs area, and for this reason it does not generate false positives. Also, this algorithms does not quickly absorb foreground, thus the bag is not included in the background and it is detected as foreground.

A further observation in the light of the results in Table 6.2 is that best performance for easy sequences is achieved with High Gradient classifier, while in case of medium and hard sequences it is achieved with Color Histogram classifier. As High Gradient algorithm is only considering color information, and High Gradient is considering edges information, the fact that one of them performs better than the other has to do with the nature of the object itself (color, size, texture...) and with its environment nature.

Figure 6.2 shows two examples of the system running for the same sequence VISOR_00, classified as “easy”, where two parked vehicles are considered as “abandoned”. Both tests have been made with PAWCS background subtraction, Subsampling algorithm for stationary foreground detection and HOG people detector, however set of images on top makes use of Color Histogram classifier while bottom set of images makes use of High Gradient classifier.

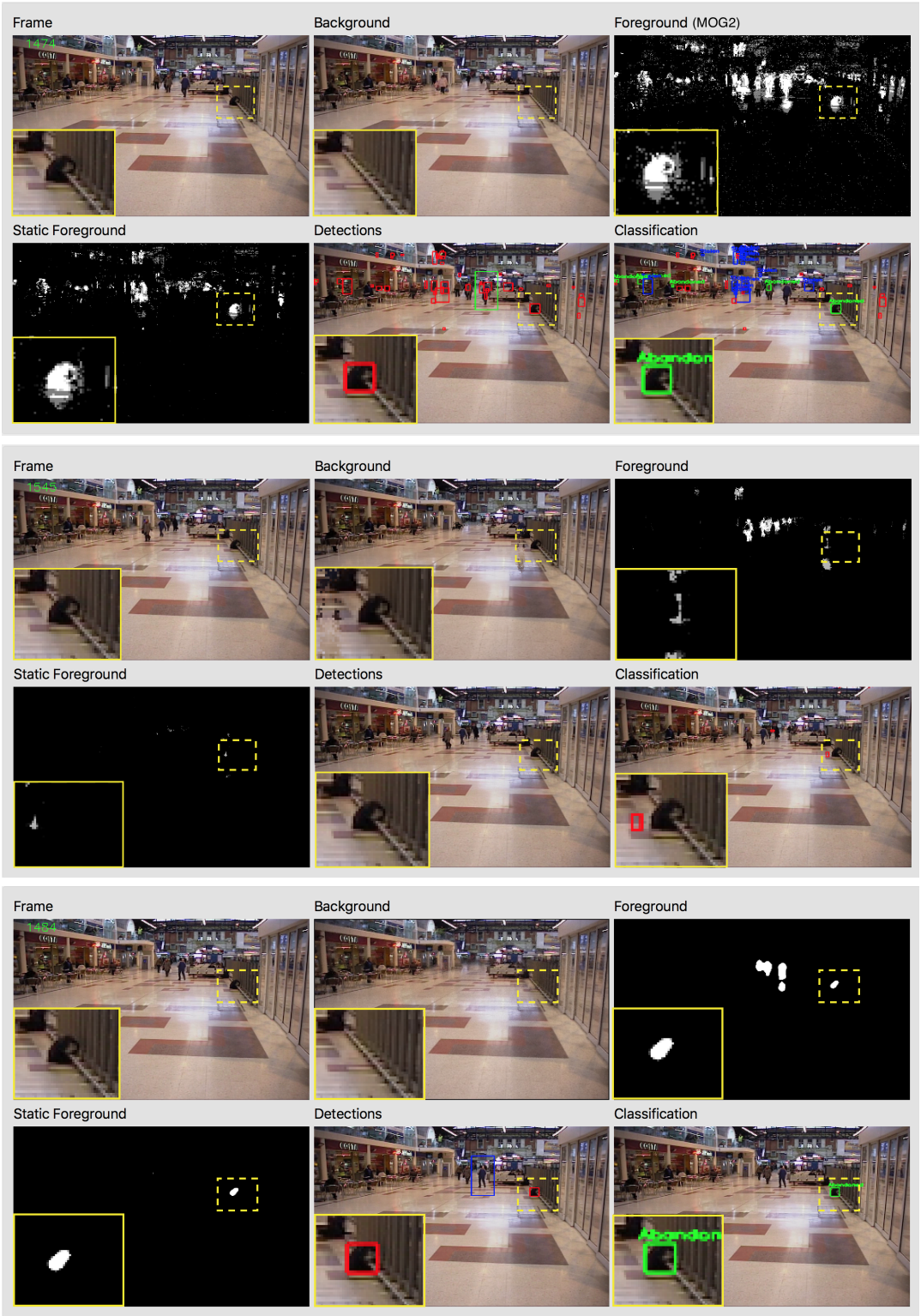


Figure 6.1: From top to bottom, sets of images tests with MOG2, IMBS and PAWCS background subtraction algorithms for the same video sequences and frame. All figures have a zoom are in the bottom left corner of the event of interest, marked in yellow. In classification display, stolen objects are marked in blue, abandoned objects in green and undefined in red.

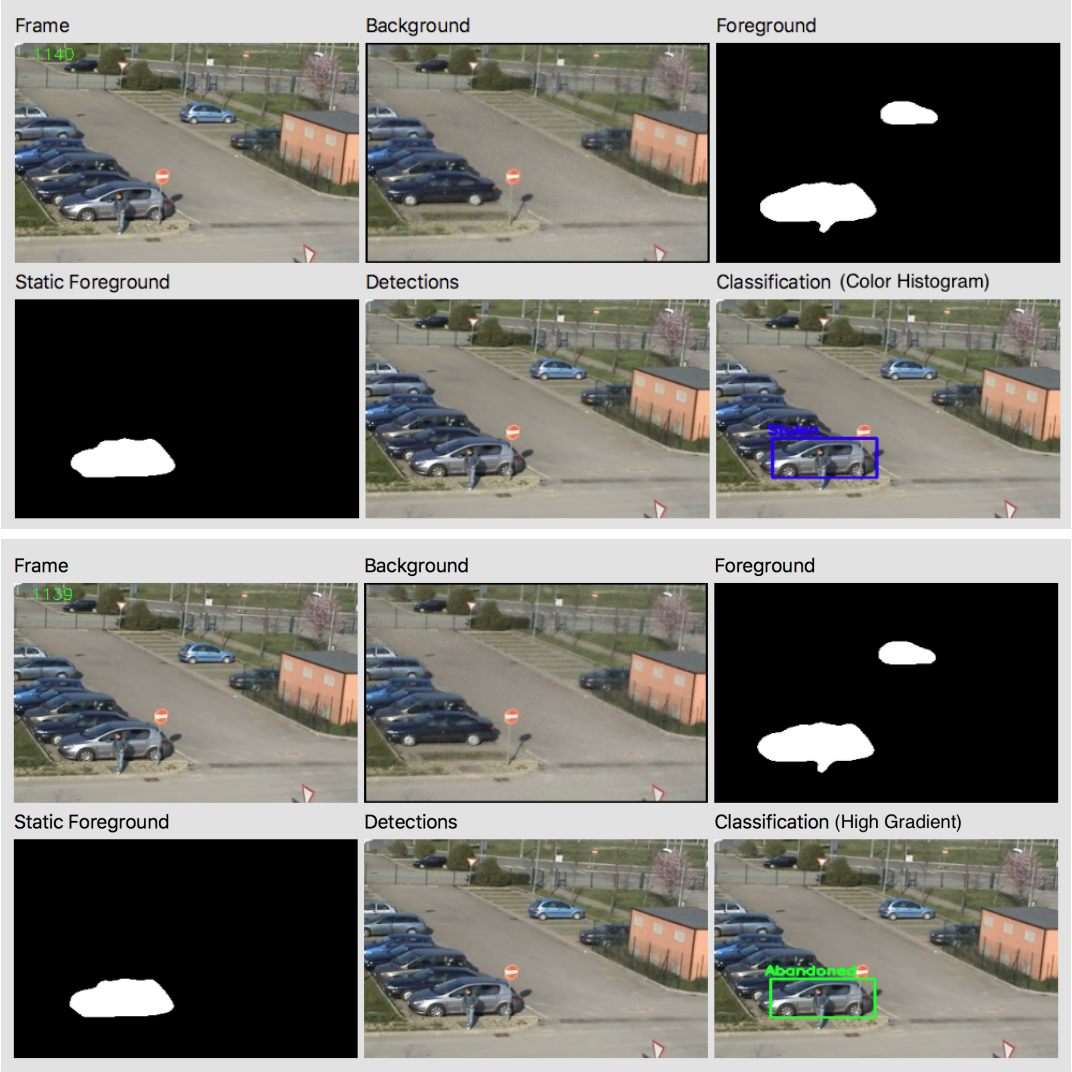


Figure 6.2: Tests over VISOR_00 sequence. Top set of images is using Color Histogram in classification module, while bottom set of images is using High Gradient classifier.

As explained in Section 2.5.2, Color Histogram is based on analysing the color information of the internal and external regions demarcated by the bounding box and the boundaries of the static object. The assumption this algorithms mades is that stolen objects in current frame present internal and external color histograms more similar than in the background model.

Looking at the top set of images in Figure 6.2, focusing on the gray car, Color Histogram algorithm is giving a false stolen detection because in this case, due to the presence of the other car behind it, color histograms of internal and external regions in the current frame are more similar that in the background.

Sequence	P	R	F	TP	FP	FN	#Abandoned objects
AVSS AB 2007	0	0	0	0	1	6	6
AVSS PV 2007	4	85	7.64	6	240	1	7

Table 6.3: Configurable system results for long-term video sequences with parameter $timeToStatic = 10$ seconds and C2 configuration.

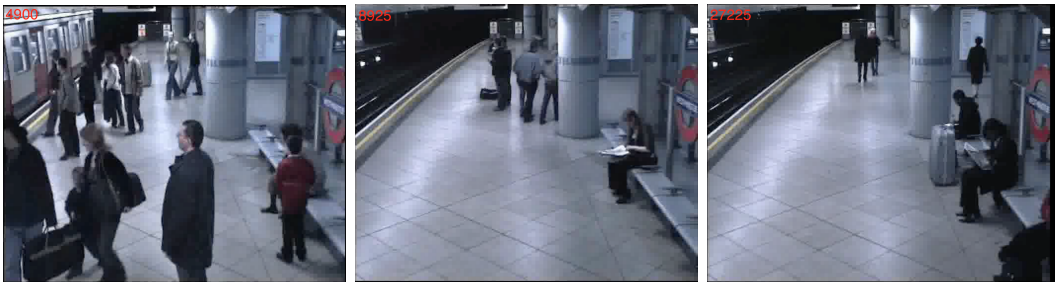


Figure 6.3: Frames number 4900, 8925 and 27225 of AVSS AB 2007 sequence showing abandoned objects.

Focusing now on the images at the bottom, High Gradient, as it takes into account information of all the car edges, it is able to detect it correctly as abandoned when comparing them to the edges in the background image.

6.1.3 Long-term

As PAWCS is the background subtraction algorithm performing better and Color Histogram is working better than High Gradient in a higher number of sequences, both algorithms, along with Subsampling for stationary foreground detection and HOG for people detection, this configuration has been used for evaluating long-term video sequences, i.e. configuration C2. Table 6.3 shows the results obtained for both sequences.

Focusing on the first row of the table, the selected configuration is not able to detect any event of interest of sequence AVSS AB 2007. In order to understand this performance, let us review the sequence. Figure 6.3, shows three examples of abandoned objects in this sequence. These three objects present hard occlusions and they are almost always surrounded by people, thus they are hard to detect. These conditions are repeating for all events in this sequence and as the abandoned object definition we have assumed filters detections when people is around the object, the system is detecting no events in this video sequence. The abandoned object definition considered in this work assumes that people around the object is the owner of it, but as has been demonstrated here, it is not always true.

Let us now analyse more in detail this events. Figure 6.4 illustrates an abandoned luggage scenario. The upper left picture is showing frame 4500, where one can observe a group of people standing on the platform. During the time these people are standing, one of them left the luggage on the floor, although it is occluded by the rest of them, in upper left image of Figure 6.4 the suitcase position is marked by the yellow dotted square. The bottom left picture shows frame 5075, where the abandoned suitcase is visible once the people occluding it move away. However, although the suitcase is completely visible it is not being detected as foreground, see bottom right picture, in consequence it is not detected as an abandoned object.

Figure 6.4: Abandoned object detection failure in long-term sequence AVSS AB 2007. Left images shows a visualization of the current frame and left images shows the foreground mask extracted with PAWCS algorithm. First and second rows correspond to frames number 4500 and 5075, respectively.

To understand the reasoning under this issue it is necessary to previously understand how PAWCS works. PAWCS algorithm is based on the characterization of background representations using a word-based approach, i.e. it registers the appearances of pixels over time as "background words" in local dictionaries using color and

texture information [51]. In conclusion, during the time people is standing occluding the suitcase, approximately 25 seconds, PAWCS algorithms is learning them as background and once they move away due to the fact that the suitcase is very similar in color to the man occluding it, it is not detected as foreground.

Another similar example where the algorithm is failing detection an abandoned bag is shown in Figure 6.5.

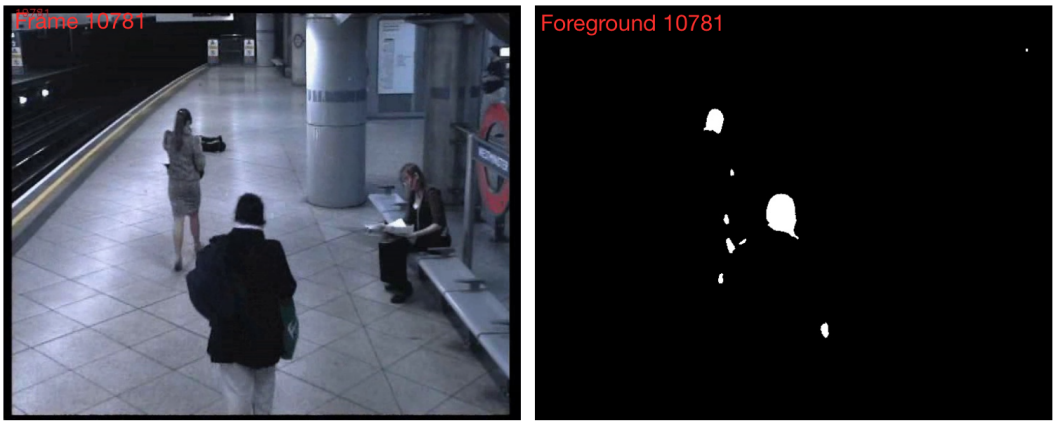


Figure 6.5: Example of the PAWCS algorithm not functioning well when detecting an abandoned bag. Left image shows the frame number 10871 and right image shows its corresponding foreground mask

Focusing now on second row of Table 6.3, one can observe that results for AVSS PV 2007 video sequence are better than the obtained for the other sequence. Although the overall results are not ideal, having a F-score measure of 7.64%, it is noticeable that Recall is quite good (85 %). The reason overall measure is that low is Precision is really low (4%). In order to understand the obtained results, let us review the AVSS PV 2007 video sequence. Whereas previous sequence was very stable, without strong changes or camera movement, this sequence present a very unstable nature. Such example is shown in Figure 6.6. This figure consists of three frames of the sequence very close in time, first and second frame are separated by only three minutes, while six minutes are the temporal difference between second and third one. Even though the little difference in time between them, they present rather different lighting conditions. The sequence passes from a full day light scene (left image) to a darker one (middle image) and again a sun-drenched scenario with strong shadows (right image). It is foreseeable that these illumination changes influence the algorithm precision. It is also important to remark that due to the position and situation of the traffic camera it is very susceptible to wind, consequently, it presents a very significant jitter along the video sequence.

Figure 6.6: Examples frames of AVSS PV 2007

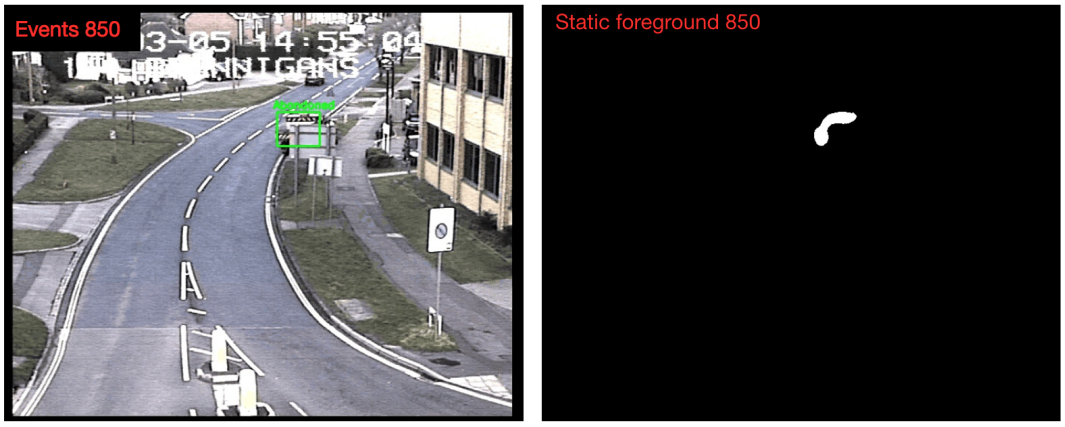


Figure 6.7: Example of abandoned vehicle detection in AVSS PV 2007 sequence. Frame with detected events and static foreground mask for frame 850 are shown in left and right images, respectively.

Let us now see some specific examples. At the beginning of the sequence a truck is parked under a traffic signal, see Figure 6.7. Up until that moment the sequence is stable with no hard jitter or illumination changes, thus the truck is correctly detected by the algorithm, as can be seen in the left image. Stationary foreground mask computed by the algorithm is shown in the right image.

Moving further in the sequence, instability becomes more noticeable. Figure 6.8 shows an instant when an orange truck is parked. Looking at left image, one can observe that the system is able to detect correctly the truck, however it also detects many false positives within the scene. Detections in the road are due to the continuous cars movement in the road, whom are detected as stationary foreground regions in the long run. There are also two false stolen detections in the building facade, that are probably due to illumination changes.

Another example detections are shown in Figure 6.9. Top images shows ground truth and events detections for frame 19150.

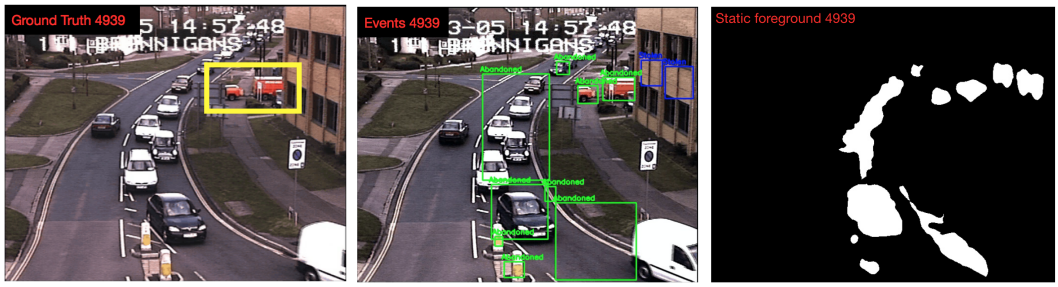


Figure 6.8: Example of abandoned vehicle detection (orange truck) in AVSS PV 2007 sequence. From left to right, ground truth, frame with detected events and static foreground mask for frame 4949 are shown.

Figure 6.9: Detected events in frames 19150 and 19993 in AVSS PV 2007 sequence. Top images show ground truth, marked with a yellow square, and detected events for frame 19150, where it is showed a correct detection of the white van. Bottom images show a misdetection of the black car.

In this frame, the white van parked on the street is correctly detected, however

the system is providing false detections on the street. Bottom images show ground truth and events detected in frame 19993. In this case, the system is failing while detecting the car and again it provides several false positives.

	E (Easy)			M (Medium)			H (Hard)		
	P	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**
Proposed System	40.50	100	57.65	18	32.14	23.08	10.29	34.29	15.82
C1	81.67	100	89.91	25.14	50	33.46	6.71	28.57	10.87
C2	73.33	88.83	80.34	26.86	64.29	37.89	9.57	41.43	15.55
C3	8.50	88.83	15.52	3.57	50	6.67	1.57	45.71	3.04
C4	12.167	61	20.29	4.86	21.43	7.912	1.57	14.29	2.83

Table 6.4: Proposed system results for short-term video sequences with parameter $timeToStatic = 10$ seconds in comparison with the configurable system results.

	AVSS AB 2007							AVSS PV 2007						
Sequence	**P**	**R**	**F**	**TP**	**FP**	**FN**	**AB**	**P**	**R**	**F**	**TP**	**FP**	**FN**	**AB**
Proposed System	2	66	3.88	4	635	2	6	4	100	7.90	7	682	0	7
C2	0	0	0	0	1	6	6	4	85	7.64	6	240	1	7

Table 6.5: Configurable system results for long-term video sequences with parameter $timeToStatic = 10$ seconds in comparison with configurable system results.

6.2 Proposed system accuracy

In order to evaluate the performance of the proposed system, the same twenty short-term and two long-term sequences evaluated in the previous section with the configurable system have been evaluated with the proposed system. Table 6.4 shows the average results, in terms of precision (P), recall (R) and F-score (F), according to the complexity (easy, medium or hard) for short-term videos in comparison with the configurable system results and Table 6.5 shows the results obtained for long-term sequences compared with the configurable system results as well.

Contrary to what one might expect, the obtained short-term overall results are not improving best short-term previous system performance configurations (C1 and C2) for easy and medium sequences, although it does improve slightly the overall results for hard sequences. However the results are better than the ones obtained with configurations C3 and C4. On the other hand, focusing on long-term results the proposed system outperforms configurable system results. Although overall performance,

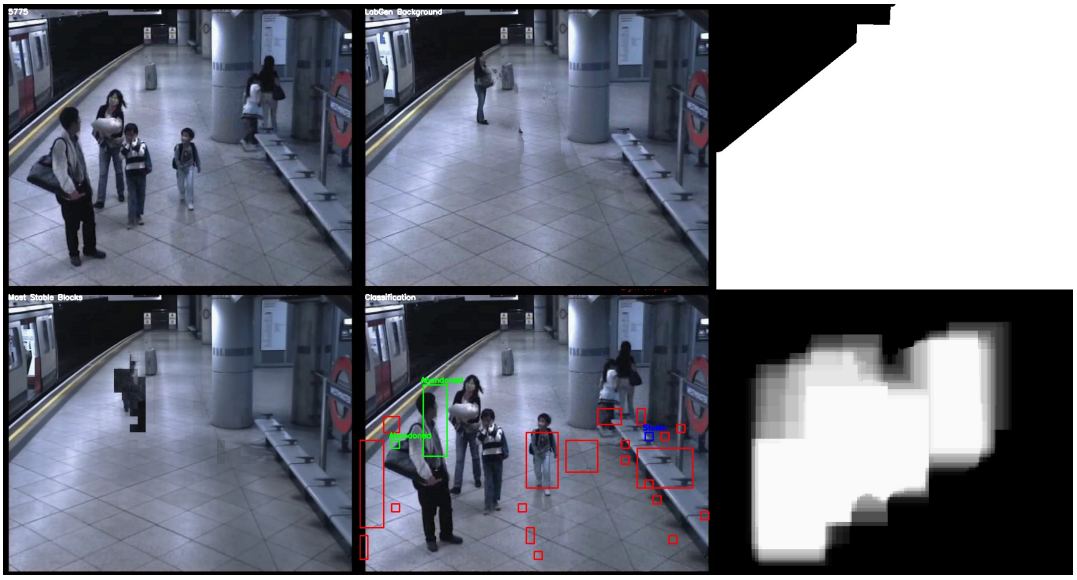


Figure 6.10: Running example of proposed system. In first row starting from left, frame, LabGen background and context mask are shown. Second row shows, from left to right, most stable blocks, detections and people detection mask.

F-score, is not really high, Recall is quite good in both sequences.

Figure 6.10 shows a running example of the proposed system with AVSS AB 2007 sequence. The event under analysis is the same shown in Figure 6.4. The proposed system, as well as the configurable system, is failing detecting the suitcase due to the occlusions it suffers and it provides false detections in the people of the scene. As this system is supposed to filter detections caused by people, we may think that somehow it is not working properly. People detection mask shown in bottom right image seems to be computed correctly, thus the system must be failing at some point of the filtering process.

Proposed system is providing a perfect Recall for AVSS PV 2007 sequence, despite of its low Precision. Let us analyse these results with some graphical examples. Left image of Figure 6.11 shows same event than Figure 6.7. Although it is not precise, the proposed system is giving a detection in the parked truck, however lots of false detections are provided within the scene. The reason this is happening is that the proposed algorithm based on spatio-temporal changes detection is very sensitive to jitter and this video sequences presents a quite hard movement of the camera that is making the algorithm to fail. Something similar happens in right image of Figure 6.11, where the black car, same as Figure 6.9, is being detected in a fairly precisely way. As before, system is detecting false events within the scene due to illumination changes and hard camera jitter.

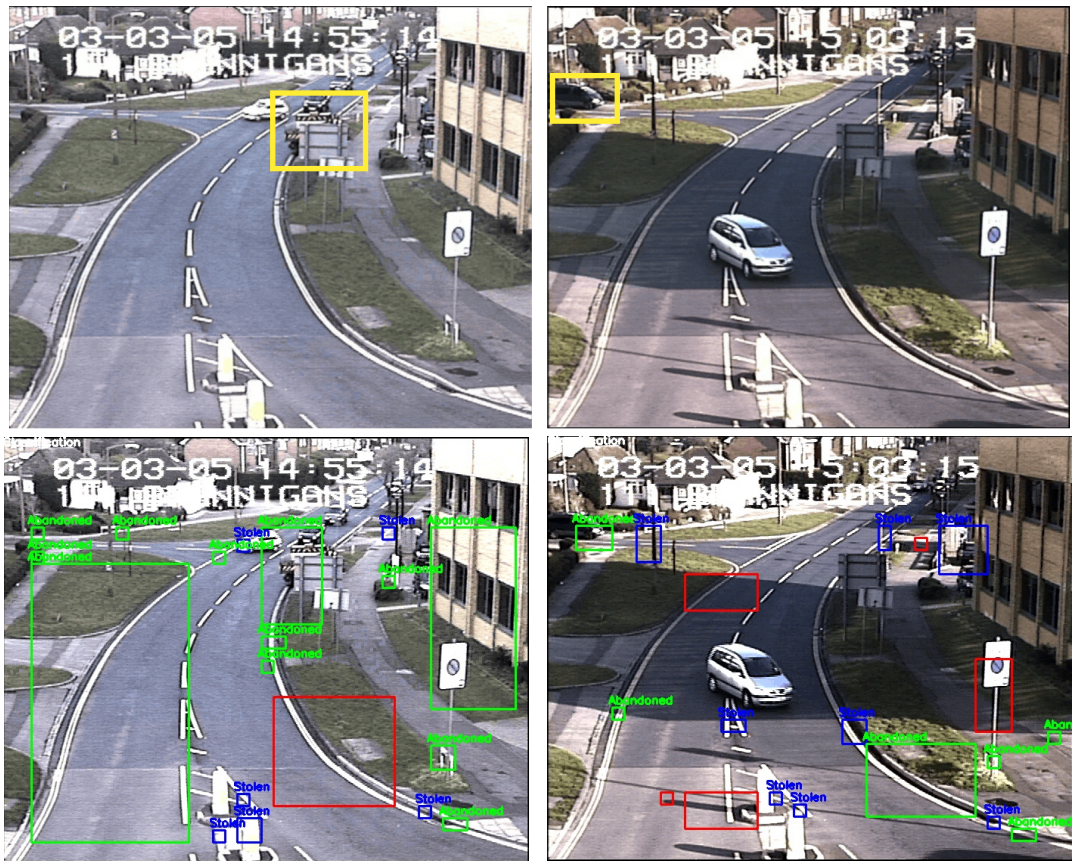


Figure 6.11: First row images show ground truth detections in both frames of in AVSS PV 2007 sequence, while second row images show detected events with the proposed system in same both frames.

6.3 Computational time comparison

This section will provide a computational time comparison of the implemented systems. All the test have been computed over $320x256$ resolution video sequences.

Figure 6.12 shows four box plots representations. These box plots depict the computational time, in terms of milliseconds per frame, of the configurable system with the four possible configurations. Looking to Sub-Figures 6.12a and 6.12b, configurations C1 and C2 respectively, it is noted that both take same computing time. It therefore follows that changing classifier algorithms from High Gradient to Color Histogram does not affect the overall time. It can be appreciated, comparing C1 and C2 with C3 and C4, that these configurations are those who are taking more computing time. Also it is important to note that background subtraction and people detection modules are the most dominant modules. Focusing on Sub-Figures 6.12c and 6.12d,

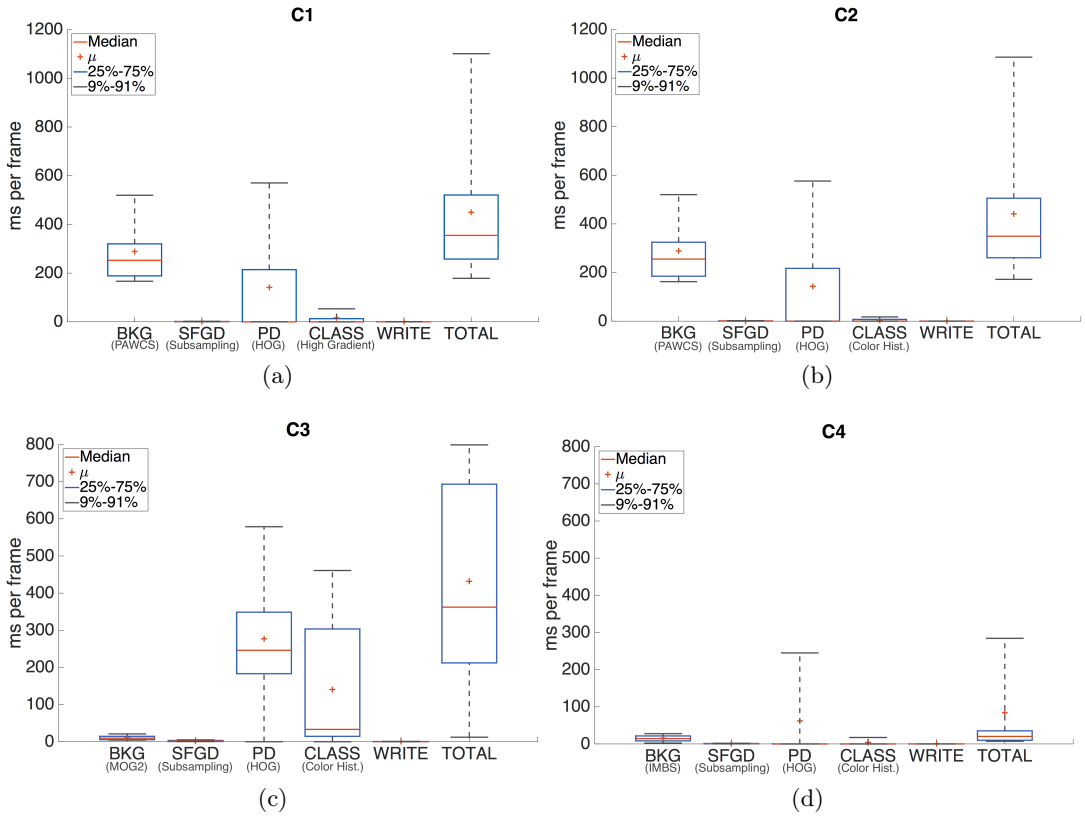


Figure 6.12: Box plots showing computational cost per module, in terms of milliseconds per frame, for each configuration of the configurable system.

it is seen that these configurations are faster than the previous ones, however they provide worst results.

Figure 6.13 shows box plot comparison between total computational time of the four configurations of the configurable system and the proposed one, where one can observe that best configuration (C2) computes, in average, 2 frames per second, while proposed system is much faster computing 10 frames per second.

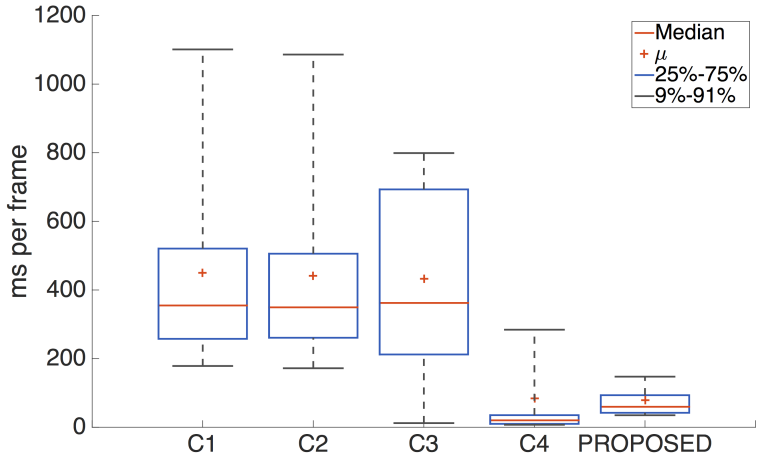


Figure 6.13: Box plot showing total computational time, in terms of milliseconds per frame, for each configurations of the configurable system and the proposed one.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

The main objective of this work was to develop an end-to-end system for abandoned and stolen object detection which would be thoroughly evaluated. For this purpose, state of the art available techniques within this field have been studied.

A complete configurable system integrating different state of the art algorithms in each module of the system has been designed and developed. In order to evaluate the performance of the system an evaluation protocol has been proposed. Twenty short-term public available video sequences have been classified into three complexity categories and the evaluation metrics and protocol have been defined. Four configurations have been selected from all possible algorithms combinations to test the performance of the system. Regarding short-term results, combination obtaining the best performance is the one made up of PAWCS algorithm for background subtraction, Subsampling algorithm for stationary foreground detection, Color Histogram for abandoned and stolen classification and HOG people detector. Although this configuration provides 89.91% F-score measure with easy sequences, results are not that good with medium and hard sequences, 33.46% and 10.87%, respectively. This reduction in performance is due to occlusions, camouflage and situations the system is not able to deal with.

In order to evaluate long-term data two public available sequences have been considered. The performance for long-term sequences has been obtained with the configuration providing the best results for short-term. For AVSS AB sequence, the system was not able to detect any of the objects. In this case PAWCS algorithm fails detecting foreground objects due to the difficulty of the sequences. For AVSS PV sequence, the system provides a good Recall (85%), although the precision was quite

low, 7.64%. The reason the precision is such low is due to the video nature itself, since it presents hard camera jitter and strong and sudden illumination changes that the system is not able to manage.

In short, after evaluating short and long-term video it might be concluded that there are three key factors in abandoned and stolen objects detection: background subtraction, abandoned/stolen definition and abandoned/stolen classifier. Background subtraction is the most critical and important stage of the process because everything else depends on it. It is very important to consider a good abandoned and stolen definition to avoid problems such as missed detections. Regarding abandoned/stolen classifiers, they are very dependent on the sequence nature itself, thus it should be chosen according to the video sequence.

In addition, a graphical user interface has been designed and developed allowing the algorithms and parameters selection and adjustment for each stage of the system, as well as displaying the results.

With the purpose of improving these results another abandoned and stolen object detection system has been proposed. The proposed system is based on detecting spatio-temporal changes and makes use of LaBGen for background computation. It has also been tested over the same short and long-term video sequences than the previous one, however, unfortunately, the system has not achieved the results that were expected. Short-term results obtained are lower than the ones obtained with the configurable system and long-term results, although they present a better overall performance, the proposed system is not able to deal with difficulties like occlusions and hard camera jitter presented in the video sequences.

7.2 Future Work

As future work aiming to continue and improve the obtained results, the following tasks are proposed regarding the configurable system:

- To increase the number of sequences to evaluate, in order to obtain a better understanding of state-of-the-art performance. For this purpose, more available short-term sequences can be considered and regarding long-term video sequences, due to the lack of available sequences, recording new long-term video sequences is recommended.

- To include more state of the art techniques to the configurable system modules in order to improve the performance.

- To test configurations, i.e. algorithm combinations, that have not been tested in the configurable system.
- To consider changing abandoned and stolen objects definition by including interaction between the owner and the object.

Regarding the proposed system, the following tasks are proposed regarding the configurable system are proposed:

- To improve filtering module by solving the problems encountered.
- To add a previous video stabilizer module in order to solve problems caused by camera jitter.

Bibliography

[1] K. N. Plataniotis and C. S. Regazzoni, "Visual-centric surveillance networks and services," *IEEE Signal Processing Magazine*, vol. 22, no. 2, pp. 12–15, 2005.

[2] C. Sacchi and C. S. Regazzoni, "A Distributed Surveillance System for Detection of Abandoned Objects in Unmanned Railway Environments," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 5, pp. 2013–2026, 2000.

[3] M. D. Beynon, D. J. Van Hook, M. Seibert, A. Peacock, and D. Dudgeon, "Detecting abandoned packages in a multi-camera video surveillance system," *in IEEE Conference on Advanced Video and Signal Based Surveillance, AVSS 2003*, pp. 221–228, 2003.

[4] N. Bird, S. Atev, N. Caramelli, R. Martin, O. Masoud, and N. Papanikolopoulos, "Real Time , Online Detection of Abandoned Objects in Public Areas," in *IEEE International Conference on Robotics and Automation (ICRA)*, no. May, pp. 3775–3780, 2006.

[5] A. Turolla, L. Marchesotti, and C. S. Regazzoni, "Multicamera object tracking in video surveillance applications," *IEE Proceedings on Target Tracking 2004: Algorithms and Applications*, pp. 85–90, 2004.

[6] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[7] Y. Tian, S. Member, and R. Feris, "Robust Detection of Abandoned and Removed Objects in Complex Surveillance Videos," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Systems*, vol. 41, no. 5, pp. 565–576, 2011.

[8] R. K. Tripathi, A. S. Jalal, and C. Bhatnagar, "A framework for abandoned object detection from video surveillance," in *National Conference on Computer Vi-*

sion, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), pp. 1–4, 2013.

[9] C.-S. Fan, J.-M. Liang, Y.-T. Lin, K.-R. Wu, K.-Y. Li, T.-Y. Lin, and Y.-C. Tseng, "A survey of intelligent video surveillance systems: History, applications and future," *Frontiers in Artificial Intelligence and Applications*, vol. 274, pp. 1479–1488, 2015.

[10] M. Valera and S. Velastin, "A review of the state-of-the-art in distributed surveillance systems," *Intelligent Distributed Video Surveillance Systems. IEE Professional Applications of Computing Series*, vol. 5, pp. 1–30, 2006.

[11] A. Bayona, J. C. SanMiguel, and J. Martínez, "Stationary foreground detection using background subtraction and temporal difference in video surveillance," *in IEEE International Conference on Image Processing (ICIP)*, pp. 4657–4660, 2010.

[12] D. Ortego and J. C. SanMiguel, "Stationary foreground detection for video-surveillance based on foreground and motion history images," *in IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2013*, pp. 75–80, 2013.

[13] G. Szwoch, "Extraction of stable foreground image regions for unattended luggage detection," *Multimedia Tools and Applications*, vol. 75, no. 2, pp. 761–786, 2016.

[14] K. Muchtar, C.-Y. Y. Lin, and C.-H. H. Yeh, "Grabcut-based abandoned object detection," *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, pp. 1–6, sep 2014.

[15] Y. Tian, M. Lu, and a. Hampapur, "Robust and efficient foreground analysis for real-time video surveillance," *Computer Vision and Pattern Recognition*, vol. 1, pp. 1182–1187, 2005.

[16] R. Wang, F. Bunyak, G. Seetharaman, and K. Palaniappan, "Static and Moving Object Detection Using Flux Tensor with Split Gaussian Models," *IEEE Change Detection Workshop (CVPR)*, pp. 414–418, 2014.

[17] J. Kim and D. Kim, "Static region classification using hierarchical finite state machine," in *IEEE International Conference on Image Processing (ICIP)*, pp. 2358–2362, 2014.

[18] J. Kim and B. Kang, "Nonparametric state machine with multiple features for abnormal object classification," *in IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS 2014)*, pp. 199–203, 2014.

[19] J. Kim, A. R. Rivera, B. Ryu, K. Ahn, and O. Chae, "Unattended object detection based on edge-segment distributions," in *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 283–288, 2014.

[20] J. Kim, M. Murshed, A. R. Rivera, and O. Chae, "Background modelling using edge-segment distributions," *International Journal of Advanced Robotic Systems*, vol. 10, no. 2, p. 109, 2013.

[21] M. Montazeri, "A self-organizing approach to background subtraction for visual surveillance applications," vol. 17, no. 7, pp. 1168–1177, 2015.

[22] L. Maddalena and A. Petrosino, "Stopped object detection by learning foreground model in videos," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 5, pp. 723–735, 2013.

[23] L. Maddalena and A. Petrosino, "The SOBS algorithm: What are the limits?," *in IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 21–26, 2012.

[24] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," *in European Conference Computer Vision (ECCV), part of Lecture Notes in Computer Science, vol 1843*, pp. 751–767, 2000.

[25] C. Cuevas, R. Martínez, D. Berjón, and N. García, "Detection of Stationary Foreground Objects Using Multiple Nonparametric Background-Foreground Models on a Finite State Machine," *IEEE Transactions on Image Processing*, vol. 26, no. 3, pp. 1127–1142, 2017.

[26] J. Pan, Q. Fan, and S. Pankanti, "Robust abandoned object detection using region-level analysis," in *IEEE International Conference on Image Processing (ICIP)*, pp. 3597–3600, 2011.

[27] K. Lin, S.-c. Chen, C.-s. Chen, D.-t. Lin, Y.-p. Hung, and A. R. Works, "Abandoned Object Detection via Temporal Consistency Modeling and Back-Tracing Verification for Visual Surveillance," *IEEE Transactions on Information Forensics and Security*, pp. 1–12, 2015.

[28] W. Wahyono and K.-H. Jo, "Cumulative Dual Foreground Differences For Illegally Parked Vehicles Detection," *IEEE Transactions on Industrial Informatics*, vol. 3203, no. c, pp. 1–1, 2017.

[29] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *Journal of the ACM (JACM)*, vol. 58, no. 3, p. 11, 2011.

[30] J. C. San Miguel and J. M. Martínez, "Robust unattended and stolen object detection by fusing simple algorithms," in *IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance, AVSSS 2008.*, pp. 18–25, 2008.

[31] J. Martínez-del Rincón, J. E. Herrero-Jaraba, J. R. Gómez, and C. Orrite-Urunuela, "Automatic left luggage detection and tracking using multi-camera ukf," in *9th IEEE International Workshop on Performance Evaluation in Tracking and Surveillance (PETS 2006)*, 2006.

[32] D. Ortego and J. C. SanMiguel, "Multi-feature stationary foreground detection for crowded video-surveillance," in *IEEE International Conference on Image Processing (ICIP), 2014*, pp. 2403–2407, 2014.

[33] Á. Bayona, J. C. SanMiguel, and J. M. Martínez, "Comparative evaluation of stationary foreground object detection algorithms based on background subtraction techniques," in *Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2009.*, pp. 25–30, 2009.

[34] H.-H. Liao, J.-Y. Chang, and L.-G. Chen, "A localized approach to abandoned luggage detection with foreground-mask sampling," in *IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance, AVSS 2008.*, pp. 132–139, 2008.

[35] S. Guler and M. K. Farrow, "Abandoned object detection in crowded places," in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS 2006)*, pp. 18–23, Citeseer, 2006.

[36] F. Porikli, Y. Ivanov, and T. Haga, "Robust abandoned object detection using dual foregrounds," *EURASIP Journal on Advances in Signal Processing*, p. 30, 2008.

[37] M. Valera and S. A. Velastin, "Intelligent distributed surveillance systems: a review," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 152, no. 2, pp. 192–204, 2005.

[38] Á. García-Martín and J. M. Martínez, "People detection in surveillance: classification and evaluation," *IET Computer Vision*, vol. 9, no. 5, pp. 779–788, 2015.

[39] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.

[40] X. Cui, Y. Liu, S. Shan, X. Chen, and W. Gao, "3d haar-like features for pedestrian detection," in *IEEE International Conference on Multimedia and Expo, 2007*, pp. 1263–1266, 2007.

[41] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005)*, vol. 1, pp. 886–893, 2005.

[42] Wahyono, J. Hariyono, and K.-H. Jo, "Body part boosting model for carried baggage detection and classification," *Neurocomputing*, vol. 228, no. October 2016, pp. 106–118, 2017.

[43] B. Wu and R. Nevatia, "Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors," *International Journal of Computer Vision*, vol. 75, no. 2, pp. 247–266, 2007.

[44] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532–1545, 2014.

[45] L. C. Campos, J. C. SanMiguel, and J. M. Martínez, "Discrimination of abandoned and stolen object based on active contours," in *8th IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS 2011)*, pp. 101–106, 2011.

[46] P. L. Venetianer, Z. Zhang, W. Yin, and A. J. Lipton, "Stationary target detection using the objectvideo surveillance system," in *IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS 2007)*, pp. 242–247, 2007.

[47] J. Connell, A. W. Senior, A. Hampapur, Y.-L. Tian, L. Brown, and S. Pankanti, "Detection and tracking in the ibm peoplevision system," in *in IEEE International Conference on Multimedia and Expo (ICME 2004)*, vol. 2, pp. 1403–1406, 2004.

[48] S. Ferrando, G. Gera, and C. Regazzoni, "Classification of unattended and stolen objects in video-surveillance system," in *IEEE International Conference on Video and Signal Based Surveillance (AVSS 2006)*, pp. 21–21, 2006.

[49] V. Gomez, "Integración y evaluación de sistemas de robo-abandono de objetos en vídeo-seguridad," proyecto fin de carrera, Universidad Autónoma de Madrid, 2016.

[50] P.-L. St-Charles and G.-A. Bilodeau, "Improving background subtraction using local binary similarity patterns," in *IEEE Winter Conference on Applications of Computer Vision (WACV 2014)*, pp. 509–515, 2014.

[51] P.-L. St-Charles, G.-A. Bilodeau, and R. Bergevin, "A self-adjusting approach to change detection based on background word consensus," in *IEEE Winter Conference on Applications of Computer Vision (WACV 2015)*, pp. 990–997, 2015.

[52] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *17th International Conference on Pattern Recognition (ICPR 2004)*, vol. 2, pp. 28–31, 2004.

[53] Z. Zivkovic and F. Van Der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern recognition letters*, vol. 27, no. 7, pp. 773–780, 2006.

[54] D. D. Bloisi, A. Grillo, A. Pennisi, L. Iocchi, and C. Passaretti, "Multi-modal background model initialization," in *International Conference on Image Analysis and Processing*, pp. 485–492, Springer, 2015.

[55] S. Bhinge, Z. Boukouvalas, Y. Levin-Schwartz, and T. Adah, "Iva for abandoned object detection: Exploiting dependence across color channels," in *in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2494–2498, 2016.

[56] D. Ortego, J. C. SanMiguel, and J. M. Martínez, "Long-term stationary object detection based on spatio-temporal change detection," *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2368–2372, 2015.

[57] B. Laugraud, S. Piérard, M. Braham, and M. Van Droogenbroeck, "Simple median-based method for stationary background generation using background subtraction algorithms," in *International Conference on Image Analysis and Processing*, pp. 477–484, Springer, 2015.

[58] L. Maddalena and A. Petrosino, "Towards benchmarking scene background initialization," in *International Conference on Image Analysis and Processing*, pp. 469–476, Springer, 2015.

[59] B. Laugraud, S. Piérard, and M. Van Droogenbroeck, "Labgen: A method based on motion detection for generating the background of a scene," *Pattern Recognition Letters*, 2016.

[60] B. Laugraud, S. Pierard, and M. Van Droogenbroeck, "Labgen-p: A pixel-level stationary background generation method based on labgen," in *International Conference on Pattern Recognition Contest Proceedings*, 2016.

[61] S. López, "Detección de objetos abandonados para vídeo-vigilancia a largo plazo," tech. rep., Universidad Autónoma de Madrid, 2018.